Treball de Fi de Grau

Grau en Enginyeria de Tecnologies Industrials

Estudi del rendiment de tècniques de mineria de dades en la predicció de resultats acadèmics

MEMÒRIA

Autora: Montse Jing Heng
Director: Luis José Talavera Méndez

Convocatòria: Gener 2019

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

RESUM

El present document tracta sobre l'anàlisi del rendiment de tècniques de mineria de dades aplicades en la predicció de l'aprovat o suspès dels estudiants de l'ETSEIB en assignatures corresponents al Q3. La mineria de dades és el procés d'extracció d'informació significativa dins d'un conjunt de dades, permetent identificar patrons i arribar a predir futures situacions. Les tècniques de predicció emprades són l'arbre de decisió i el mecanisme *Bagging* prenent l'arbre de decisió com a estimador base.

Tot el procés d'anàlisi ha estat adaptat a la metodologia CRISP (un dels models referència en la mineria de dades), des de la preparació de les dades fins la validació dels models. A partir dels resultats obtinguts en la validació s'han pogut contrastar els dos mètodes de predicció utilitzats.

Les eines utilitzades en el treball giren entorn al llenguatge de programació Python i són totes de programari lliure. Concretament, s'ha fet ús de les llibreries *Pandas* i *scikit-learn* i la distribució *Anaconda* com a IDE.

La conclusió principal que s'extreu del projecte és que les dades que es disposen són poc representatives per tal de poder ser predites. Les precisions de predicció són relativament baixes en els dos mecanismes utilitzats. Al final del treball es troba una secció de treball futur on es proposen alternatives d'estudi per tal d'aprofundir l'anàlisi.

S U M A R I

ETSEIB

GLOSSARI

En aquest glossari es definiran alguns termes d'ús regular en el treball.

DataFrame Format d'element en forma de taula amb el que treballa la llibreria *Pandas.*

Float En codi Python, format d'un element en forma de nombre decimal.

Missing values Valors que es troben a les cel·les en un *DataFrame* quan no tenen cap valor assignat

Nan *Not a number*. En codi Python, format que prenen els *missing values*, els quals són interpretats com valors buits.

Overfitting Fenomen que succeeix quan un algorisme de predicció està massa ajustat a les dades amb el que es modela.

Pandas Llibreria Python d'on provenen les funcions emprades en l'anàlisi de dades.

Python Tipus de llenguatge de programació emprat en el treball.

String En codi Python, format d'element que pot incorporar components tant alfabètics com numèrics.

Testing Fase de predicció que consisteix en la validació dels models construïts. El conjunt de dades sobre les que es valida també rep aquest nom.

Training Fase de predicció que consisteix en la construcció de models predictors. El conjunt de dades sobre els quals es modela el mecanisme rep també aquest nom.

1. INTRODUCCIÓ

Al llarg dels anys, la tecnologia s'ha anat consolidant com una part essencial de les nostres vides. La digitalització de molts processos involucrats ha permès facilitar i agilitzar tot tipus de serveis.

Aquest fenomen es veu reflectit des de procediments concrets com l'enregistrament de productes en codis de barres fins a grans processos com la gestió d'operacions en una indústria. Darrere de tots els processos es troba una recollida d'informació, en la qual la generació diària de dades és tan massiva que molts asseguren que ens trobem en l'era de les dades. Si aquestes dades es gestionen i analitzen de la forma adequada, poden donar lloc a nous coneixements.

Aquí és on neix la mineria de dades, consistent en l'estudi o anàlisi de dades mitjançant tècniques automàtiques com arbres de decisió o mètodes que involucren l'acció humana, sigui mitjançant eines gràfiques o la identificació de patrons amb l'ajuda de computadors. A partir de l'anàlisi es pot arribar a generar nous coneixements a partir dels quals es poden predir situacions futures.

Un exemple reflectit dins de la nostra vida quotidiana és el cas dels comerços, els quals posseeixen grans volums de dades generats per les nostres compres. Un anàlisi adequat de les dades recollides pot donar lloc a nova informació, com és el cas de la creació de futurs perfils de client, la qual permet optimitzar l'atenció al client i enfocar els productes a les seves necessitats. El coneixement de quins són els productes més venuts a cada temporada permet crear previsions de demanda, conegudes com a *forecasts* en l'àmbit del *supply chain* (gestió de cadenes de subministrament). Mitjançant dades d'afluència de compradors en un establiment es poden realitzar prediccions de futures afluències, fent possible una millor gestió de l'horari d'obertura o dels torns del personal.

A continuació es planteja una possible situació: imaginem un supermercat on s'enregistren totes les dades de compres. Un extens anàlisi mitjançant mineria de dades revela un tipus de patró important: els divendres i dissabtes la venta de cervesa és major de forma destacable i, conjuntament, les compres de cervesa estan acompanyades de patates fregides o olives. L'establiment podria fer ús d'aquesta informació i incitar encara més la compra de cervesa, patates fregides i olives apropant les estanteries on es venen.

El reconeixement de patrons ajuda a la definició de criteris de venta en el comerç, permetent saber quins són els aspectes que es poden millorar amb més facilitat.

La mineria de dades no està només limitada al món del negoci, àmbit on es fa un gran ús d'ella. Aquesta és aplicada també en altres àrees com la medicina per portar el registre dels diagnòstics de pacients i poder identificar les millors pràctiques per cada cas. Un cas d'exemple és el de *Mayo Clinic*, un grup de recerca i pràctica mèdica especialitzat en tractaments de prevencions terciàries com la gestió del càncer o la neurocirurgia. [1] Aquest grup ha estat col·laborant amb IBM per desenvolupar una eina virtual per identificar com els últims pacients han respòs a certs tractaments en particular.

En resum, la minera de dades és una poderosa eina en l'anàlisi de dades que aplica la tecnologia en la creació de nova informació en forma d'identificació de patrons o tendències, detecció d'anomalies i reconeixement de relacions entre factors.

Metodologia CRISP

Per tal de conduir l'anàlisi de mineria de dades de forma sistemàtica, és convenient seguir una metodologia. Hi ha diferents processos estàndards, entre els quals destaca la metodologia CRISP (*Cross-Industry Standard Process*). [2] Segons el portal online *KDnuggets*, especialitzat en ciència de dades, CRISP és la metodologia més utilitzada en projectes de mineria de dades, imposant-se amb un 43% d'ús respecte les altres metodologies.

CRISP consisteix en una seqüència d'etapes, formant un procés cíclic de sis fases diferenciades, les quals es troben representades a la *Figura 1*. Proporciona una seqüència ideal de fases, depenent del projecte a aplicar pot variar cert ordre entre fases o alguna fase pot no ser necessària. La metodologia està composta per les següents fases:

- **Comprensió del problema**: Consisteix en la base del projecte. Inclou les tasques de comprensió d'objectius i requisits del projecte per tal de convertir-los en objectius tècnics i en un pla de projecte. Comprèn la valoració de la situació i la definició

[1] N. Swartz (2015). IBM, Mayo clinic to mine medical data, *The Information Management Journal*.

[2] G. Piatetsky (2014). CRISP-DM, still the top methodology for analytics, data mining or data science projects*, KDNuggets*.

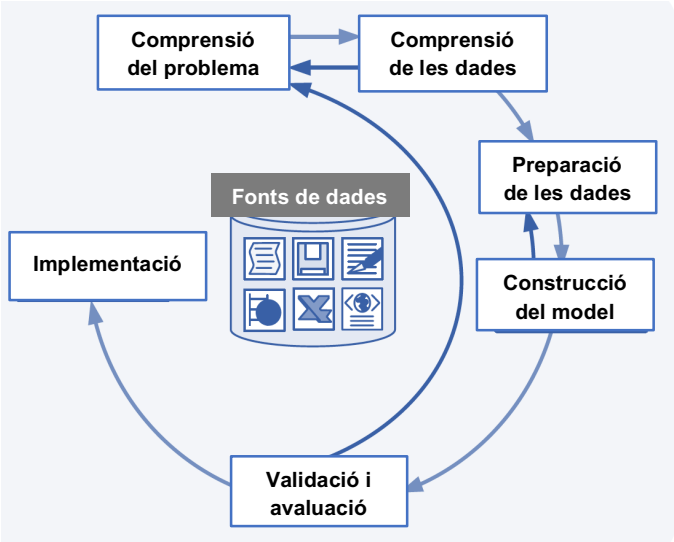

Figura 1. Metodologia CRISP

d'objectius. Aquest enteniment permet preparar les dades i interpretar els resultats de la forma correcta.

- **Comprensió de les dades**: Consisteix en la recol·lecció inicial de dades per tal de tenir un primer contacte amb el problema a resoldre. La familiarització amb les dades (descripció i exploració d'elles) porta a la verificació de la seva qualitat i la formulació de primeres hipòtesis.

- **Preparació de les dades**: Es preparen les dades per tal que s'adeqüin a les tècniques de mineria de dades que s'utilitzaran posteriorment. Això implica la selecció, neteja i transformació de dades cap al format desitjat tenint en compte les relacions entre les variables inicials.

- **Modelatge**: En aquesta fase es seleccionen les tècniques de modelatge més apropiades pel projecte, d'acord a les seves característiques per ser aplicades en les dades disposades i poder complir els objectius proposats. Un cop escollida es procedeix a la construcció del model. A la vegada, s'estableix també el pla de prova, definint el mètode d'avaluació de models per dur a terme la verificació dels resultats.

- **Validació i avaluació**: Consisteix en l'avaluació dels resultats del model construït en termes de fiabilitat i validesa mitjançant tècniques de verificació. Qualsevol deficiència detectada condueix a la revisió del procés total.

- **Implementació:** Un cop el model ha sigut construït i validat, el nou coneixement obtingut a partir de les dades es transforma en la consolidació d'una estratègia d'implementació que comporti la millora en l'àrea tractada pel projecte.

En el present treball, es pretén aplicar la mineria de dades en les dades acadèmiques de l'alumnat de l'ETSEIB, enregistrades per part de l'escola des de la implementació del nou pla d'estudis al 2010. Els objectius plantejats en un principi es presenten a continuació.

1.1. Objectius

L'estudi del present treball consisteix en l'anàlisi del rendiment de les tècniques de mineria, concretament els arbres de decisió i el mecanisme *Bagging,* en la predicció de l'aprovat o suspès de l'alumnat de grau de l'ETSEIB (*Escola Tècnica Superior d'Enginyeria Industrial de Barcelona*) en les assignatures corresponents al primer quadrimestre posterior a la fase inicial (Q3). Els objectius principals són els següents:

- **Aplicació d'una metodologia.** Tot el procés d'anàlisi de dades s'ha de dur a terme en base a una metodologia general rigorosa, la qual ha d'estar ben documentada per tal de poder ser replicada en un futur.

- **Estudi i comparació de mètodes de classificació**. Existeixen nombrosos mètodes classificatoris de predicció. D'aquests, s'estudiaran els arbres de decisió i els models *Bagging* i a partir dels resultats obtinguts, s'analitzarà el seu rendiment en funció d'uns certs paràmetres i es contrastaran els algorismes.

- **Validació de resultats de forma sistemàtica.** Per tal de validar el model de predicció construït, s'elegirà amb quin mode es durà a terme. Serà important aplicar el mateix mètode de validació en els models predictius per tal d'avaluar l'eficàcia d'aquests.

A més dels objectiu principals, es poden definir també els següents objectius complementaris:

- **Coneixement de la llibreria *Pandas.*** *Pandas* és una llibreria de Python amb la que es treballarà durant tot el projecte. Es requereix un cert grau de coneixement sobre aquesta per tal d'implementar les funcions que incorpora durant tot el codi informàtic.

- **Familiarització amb l'entorn.** Tota la programació del codi es durà a terme amb l'IDE *Anaconda*, la qual engloba totes les eines necessàries per la realització del treball. S'estudiaran quines de les eines que incorpora es podran aprofitar per acomplir les diferents fases del projecte.

1.2. Abast del projecte

Al tractar-se d'un projecte de mineria de dades, aquest seguirà la metodologia CRISP exposada a la secció introductòria. Les fases que incorpora s'han adaptat als objectius i limitacions del treball resultant en:

1. **Comprensió del problema**. Abans d'endinsar-nos a l'exploració de les dades, és indispensable conèixer el problema de fons. Com a estudiant, s'han experimentat totes les fases del grau, les diferents metodologies i l'entorn de l'escola. Es considera que el coneixement del problema és suficient per poder dur a terme el treball.

2. **Comprensió i preparació de les dades**. Donat que les variables de les dades inicials no són molt complexes, no és necessari dedicar tota una fase a la seva comprensió. A partir del coneixement de què representen les dades i variables, es procedirà a realitzar la preparació d'aquestes. Es definirà com organitzar les dades a analitzar, incloent la seva neteja i transformació. A mesura que es va definint l'estructura de dades desitjada, s'implementarà alhora en codi perquè la preparació de dades a partir de les dades inicials sigui automàtica.

3. **Modelatge i validació**. Un cop obtingudes les dades organitzades de forma òptima, es procedirà a l'estudi d'aquest mitjançant algorismes de predicció. Es seleccionaran els mètodes de predicció a utilitzar i s'analitzarà com varia l' efectivitat en funció dels paràmetres emprats. Un cop construït el model es realitzarà la validació dels resultats.

Tal i com s'observa, l'abast del projecte comprèn quasi totes les fases d'un projecte convencional de mineria de dades. L'única fase no tractada és la fase d'implementació al trobar-se limitada, donat que consistiria en posar en funcionament el mètode predictiu a l'escola.

1.3. Eines utilitzades

Donat que es tracta d'un projecte de mineria de dades, totes les eines utilitzades durant el treball són informàtiques, amb les que s'efectuaran totes les fases que comprèn l'anàlisi de dades. A continuació s'exposen les eines emprades:

Python

Python és un llenguatge de programació concebut i implementat a finals dels anys 80, el seu ús ha anat evolucionant exponencialment fins avui. És un llenguatge caracteritzat per la simplicitat i, alhora, potència que ofereix en quant a varietat de possibilitats d'implementació.

Figura 2. Python

En comparació a altres codis, permet construir estructures més entenedores al tenir un aspecte molt més visual. A més a més, inclou una gran quantitat de llibreries per tot tipus d'àrees d'estudi (moltes de les funcions necessàries ja estan programades).

Els usuaris d'altres tipus de llenguatges defensen que els usuaris de Python tendeixen a acostumar-se massa a les característiques d'aquest i els hi suposa una gran dificultat aprendre un altre tipus de llenguatge. És cert també, que s'ha demostrat que Python és un llenguatge menys potent que els altres per aplicacions de programació mòbil.

Tot i així, Python segueix sent un dels llenguatges més utilitzats, sobretot en àmbits d'intel·ligència artificial i anàlisis predictius. [3] Segons el portal online *KDnuggets*,, les enquestes realitzades entre desenvolupadors d'anàlisi de dades revelen que gran part d'ells opta per l'ús de Python i els elements relacionats amb aquest.

A la *Figura 3* es troben representats els resultats a les enquestes d'ús d'eines d'anàlisi de dades, realitzats cada any durant el període 2016-2018. Per davant de tots es troba Python, el qual ha anat creixent al llarg del temps. Entre les eines més utilitzades també es troben *Anaconda*, un entorn de desenvolupament per Python i *scikit-learn*, una llibreria Python.

S'ha escollit aquest tipus de llenguatge principalment per la gran comunitat d'usuaris que posseeix, la claredat que ofereix en la lectura de codi, la seva versatilitat i la

[3] G. Piatetsky (2018). Python eats away at R: Top Software for Analytics, Data Science, Machine Learning in 2018*, KDNuggets*.

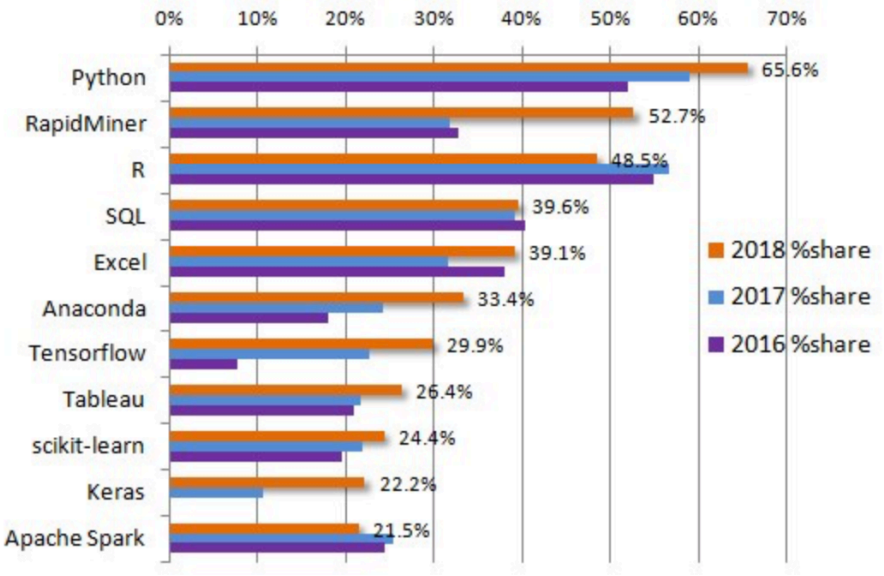

Figura 3. Eines més utilitzades en anàlisi de dades en el període 2016-2018

Figur

disposició de llibreries d'anàlisi de dades de fàcil accés. A més a més, un aspecte a considerar per l'elecció d'aquest codi ha sigut el fet de ser el llenguatge impartit en les dues assignatures d'informàtica (*Fonaments d'Informàtica* i *Informàtica*) que s'instrueixen a l'escola.

Pandas

Pandas és una llibreria Python de codi obert, la qual conté tot tipus de funcions que faciliten la manipulació de grans volums de dades. Proporciona especialment les funcions necessàries per la part de preparació de dades que s'utilitzaran en un posterior anàlisi, evitant haver de fer ús d'un altre domini. Els punts a destacar d'aquesta llibreria són els següents:

- Treball amb l'element *DataFrame*, objecte propi d'aquesta llibreria que es basa en una taula de dades que incorpora la indexació automàtica de files.
- Incorporació d'eines de lectura i escriptura de dades de formats típics com fitxers .csv i .txt, documents de càlcul com .xslx i bases de dades SQL entre altres.
- Flexibilitat en la remodelació i *pivoting* de conjunts de dades, a més de facilitats en la fusió i unió entre ells.
- Alineació de dades intel·ligent i maneig integrat de *missing values*, permet organitzar les dades en base a etiquetes i manipular dades desorganitzades de forma senzilla.
- Possibilitat d'addició o transformació de dades amb la poderosa funcionalitat de *group by*, fent possible l'aplicació de diferents operacions en conjunts de dades específics.

La part preparació de dades, tant la part de neteja com la de transformació de dades, és durà a terme amb aquesta llibreria.

Spyder (Anaconda)

Figura 4. Anaconda

Anaconda és una distribució Python formada per un gran nombre de paquets de llibreries i entorns de treball de ciència de dades. Permet la instal·lació i posada en marxa d'aplicacions i editors com *Spyder* i proporciona una interfície gràfica des de la qual es poden manipular els diferents entorns. Mitjançant la instal·lació del paquet Anaconda s'obtenen tots els elements de treball que inclou, d'entre els quals es farà un gran ús de l'IDE *Spyder* i la llibreria *Scikit-learn*.

El codi font que aplicat durant el treball s'editarà en el programari *Spyder*, un IDE (*Integrated Development Environment*) desenvolupat en Python, el qual consisteix bàsicament en una aplicació informàtica que proporciona un marc de treball en el desenvolupament del software. Els blocs principals de treball que inclou són els següents:

- **Editor** L'editor multi-llenguatge integra diferents eines per tal que l'experiència d'edició de codi sigui l'òptima. Algunes eines són l'anàlisi de codi en temps real, la qual alerta de possibles errors dins del codi, l'exploració de funcions i classes per un fàcil accés a les seves definicions i característiques o el buscador dins del codi.

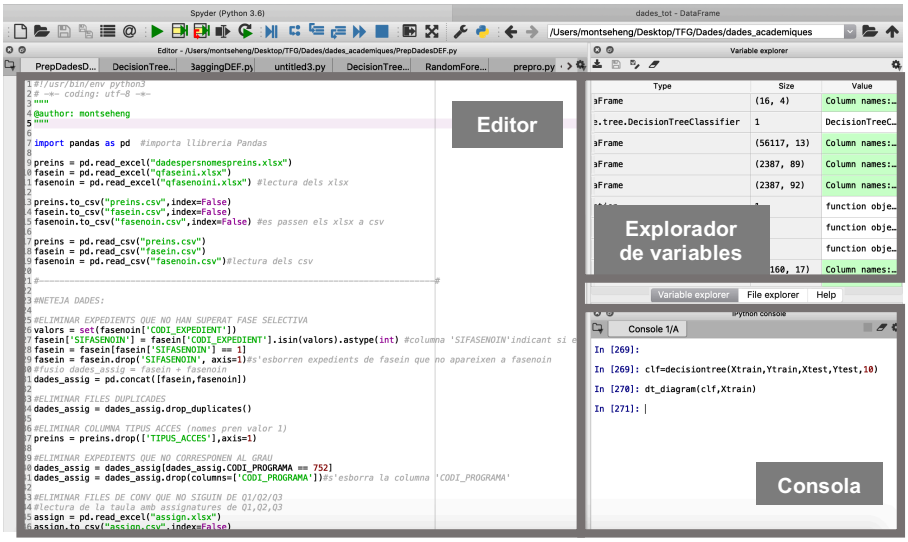

Figura 5. Finestra de l'Spyder

- **Consola** La consola integrada és concretament *IPython*, una consola que permet executar comandes i interactuar amb el sistema d'una forma gràfica. Proporciona

facilitats en la posada en marxa de fitxers i en la interrupció d'execucions sense afectar als altres elements d'*Spyder*.

- **Explorador de variables** L'explorador de variables mostra tots els elements creats en la consola durant la sessió, elements com variables, funcions o mòduls. A més de poder-los visualitzar, permet aplicar operacions com l'edició d'objectes o eliminació directa de variables.

Scikit-learn

Scikit-learn és un dels elements que venen instal·lats amb la distribució Anaconda. Es tracta d'una llibreria Python que proporciona diversos algoritmes per l'anàlisi de dades, estalviant temps en la programació de codi per la construcció del model i permetent ficar més èmfasi en el propi anàlisi. Les funcionalitats principals que abasta són:

Figura 6. Scikit-learn

- **Regressió** L'anàlisi per regressió és un conjunt de processos estadístics en què s'estima la relació entre variables, focalitzant l'estudi en la relació entre una variable dependent i una o més variables independents. Ajuda a entendre com varia el valor d'una variable depenent del valor que pren una altra variable. La llibreria inclou diferents algorismes de regressió per l'anàlisi, tant lineals com logístiques.
- **Classificació** La classificació consisteix bàsicament en identificar a quin conjunt de categories pertany una nova dada a partir d'un conjunt de dades on es coneixen de quines categories són membres, conegut com dades d'entrenament (*training set*). És un tipus de reconeixement de patrons, comportaments que es repeteixen entre el conjunt de dades.
- ***Clustering*** L'agrupament, conegut com *clustering*, consisteix en l'anàlisi de dades mitjançant l'agrupació de les dades de manera que els dades d'un mateix grup són més similars entre elles en algun aspecte que les dades d'un altre grup.
- **Selecció de model** Proporciona eines per la comparació entre diferents models i la selecció de paràmetres òptims en un model determinat.

2. COMPRENSIÓ I PREPARACIÓ DE DADES

La preparació de dades conforma la base de l'anàlisi a realitzar, formant part d'un dels determinants claus en el resultat de l'estudi. Es pot afirmar que, a més de ser una de les parts més importants, també es tracta d'una de les parts més difícils i llargues en quant a presa de decisions: la selecció o rebuig de certes dades a explorar posteriorment pot canviar de forma considerable els resultats.

2.1. Dades inicials

Es disposa de les dades acadèmiques de l'alumnat de l'ETSEIB, concretament del període comprés entre l'any 2010 i l'any 2018. Aquestes dades es distribueixen en 3 documents diferents: un corresponent a les dades de preinscripció, un corresponent a la fase inicial i un últim a la fase no inicial.

Les dades de preinscripció de l'alumnat estan registrades a partir de les següents variables:

- **CODI_EXPEDIENT**: Codi adjudicat a l'expedient d'un estudiant determinat per la gestió d'aquest.
- **SEXE**: Home (H) o dona (D).
- **CP_FAMILIA**: Codi postal de la residència familiar de l'estudiant.
- **ANY_ACCES**: Any d'accés a l'escola (any en què es formalitza la preinscripció).
- **TIPUS_ACCES**: Forma d'accés a l'escola (hi ha un únic valor indicat com a 1).
- **NOTA_ACCES**: Nota de selectivitat amb la que s'accedeix al grau.
- **CP_CENTRE_SEC**: Codi postal del centre d'educació secundària d'on prové l'estudiant.

Les dades de qualificacions, corresponents tant a la fase inicial com a la fase no inicial, s'organitzen en:

- **CODI_PROGRAMA**: Codi de l'estudi cursat. En el cas del grau, aquest està representat pel 752.
- **CODI_EXPEDIENT**: Codi adjudicat a l'expedient de l'estudiant per la seva gestió.
- **CODI_UPC**: Codi de l'assignatura a la que correspon la qualificació.
- **CREDITS**: Nombre de crèdits ECTS que computa l'assignatura.
- **CURS**: Any en què es cursa l'assignatura.

- **QUAD**: Quadrimestre en què es cursa l'assignatura (Q1 pel quadrimestre de tardor i Q2 pel quadrimestre de primavera).
- **SUPERA**: Aprovat o no aprovat de l'assignatura (S = aprovat, N = no aprovat).
- **NOTA_PROF**: Nota final de l'assignatura assignada pel professor de l'estudiant.
- **NOTA_NUM_AVAL**: Nota final assignada per l'avaluació curricular.
- **NOTA_NUM_DEF**: Nota final definitiva després del període d'avaluació curricular.
- **GRUP_CLASSE**: Grup de classe en què es cursa l'assignatura.

A partir d'aquestes dades inicials es desitja realitzar les transformacions necessàries per maximitzar el seu enteniment i facilitar l'exploració d'elles.

2.2. Preparació de dades

Tal i com s'ha mencionat anteriorment, les dades que es disposen es troben distribuïdes en 3 taules diferents. Donat que es treballa amb llibreria *Pandas*, aquestes taules s'han convertit a *DataFrame*, nom que reben el format d'aquestes taules. Tot el codi emprat en la preparació de dades es troba a l'Annex.
A continuació es descriu el procediment seguit per fusionar les dades a un únic *DataFrame*.

2.2.1. Neteja de dades

Dades duplicades o redundants

Es defineixen com a dades redundants aquelles que un cop eliminades no produeixen cap pèrdua d'informació essencial. Aquest és el cas de la variable *TIPUS_ACCES*, al tractar-se d'una columna que consta del mateix valor (1) en totes les cel·les.

Es pot donar que dos o més files posseeixin dades idèntiques accidentalment, en aquest cas ens trobaríem en un cas de dades duplicades. Aquest cas ens podria donar problemes ja que certs models de predicció tenen grans dificultats en digerir aquestes dades.

Domini

Les dades inicials contenen una gran quantitat de dades mentre que l'estudi es limita a un cert domini. És a dir, l'estudi realitzat serà aplicable només als expedients que compleixin les condicions que s'esmenten a continuació.

De totes les dades corresponents a tot l'alumnat de l'escola, es seleccionen només aquells que corresponen al grau (els que prenen valor 752 en la variable CODI_PROGRAMA), ja que són els únics possibles a estudiar seguint els objectius del treball.

Hi ha un grup d'estudiants que es troben encara en la fase inicial del grau o just han acabat la fase inicial i no han iniciat encara la fase no selectiva. Donat que per la predicció es necessita que l'alumne hagi finalitzat la fase inicial i hagi cursat ja alguna assignatura de la fase no selectiva, els casos mencionats anteriorment s'han de descartar. És a dir, es descarten aquells expedients de fase inicial que no apareixen la taula de fase no inicial.

Missing values

Missing values fa referència a aquelles cel·les que per algun motiu no tenen cap valor (estan en blanc). Un cop passades en un *DataFrame* prenen valor *nan* (*not a number*). Depenent de a quina variable es produeix aquest valor buit, té major o menor importància.

Certs mètodes de predicció accepten l'entrada de *missing values* ja que en el procés ignoren aquests valors, mentre que altres no són capaços d'ignorar-los i totes les dades a entrar han de contenir algun valor. Més endavant es trobarà el cas que hi ha algorismes que no accepten variables d'entrada de tipus *nan*, de forma que s'haurà d'optar per atribuir a aquests casos un valor en concret.

En el codi s'ha creat una variable *valorsnuls* que recol·lecta totes les files amb algun *missing value*. En els *DataFrame* estudiats s'observa que només trobem aquest tipus de valors a les variables corresponents a codis postals o notes numèriques d'assignatures.

2.2.2. Transformació de dades

Pivoting

Les dades inicials contenen a cada fila una convocatòria cursada d'una determinada assignatura per un determinat expedient. Aquest format dificulta la comprensió de la informació al ser una estructura poc visual.

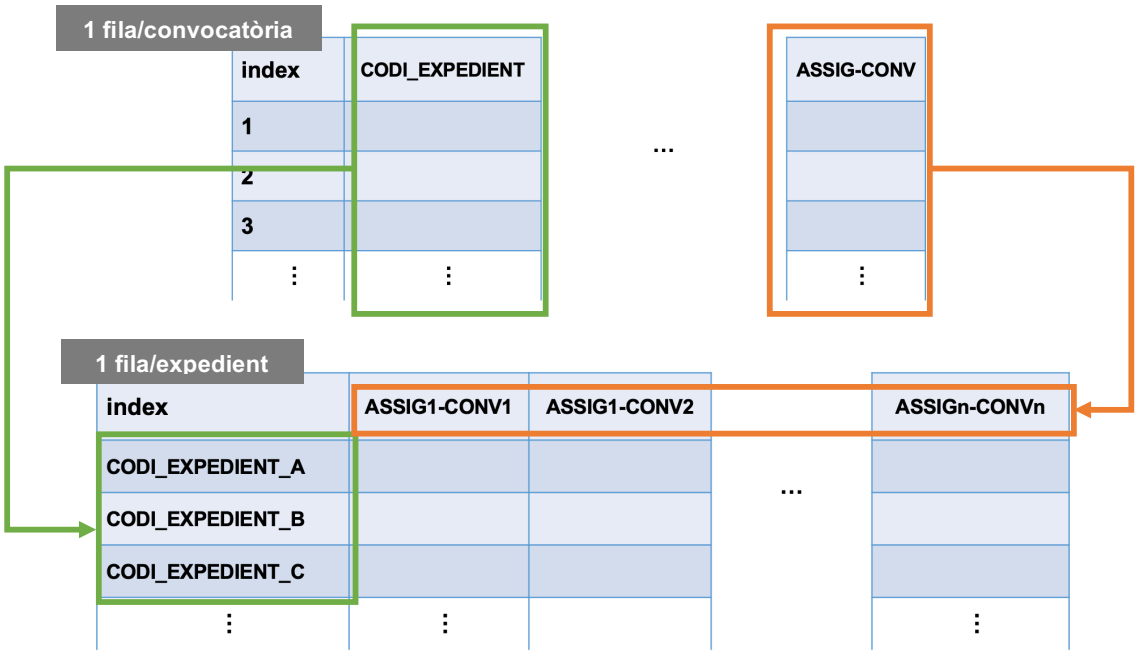

Figura 7. Procés de pivoting

Per una fàcil manipulació de les dades, es desitja traslladar les dades en una nova taula on totes les dades d'un mateix expedient es trobin en una única fila. El procés emprat està representat en la *Figura 7*. Primerament es crea una columna a la taula inicial indicant a quina combinació assignatura-convocatòria correspon cada fila. Seguidament es traslladen les dades a una nova taula on cada columna correspon a una combinació assignatura-convocatòria i cada fila correspon a un expedient concret.

La llibreria *Pandas* conté una funció (*Pandas.Dataframe.pivot)* que realitza l'acció de traslladar les dades de la taula inicial en un format com el descrit. Aquest procés rep el nom de *pivoting*, on cada fila de la taula inicial passa a ser una columna de la taula final. Aquesta taula final està guardada en la variable *dades_assig* i conté una fila per expedient, una columna per cada assignatura-convocatòria i les cel·les contenen les notes numèriques. En cas de no haver cursat l'assignatura o no haver necessitat cursar aquella convocatòria de l'assignatura, la cel·la conté un *nan*.

Fusió de dades

Al partir de tres taules diferents, un dels objectius en la preparació de dades és la seva fusió en una única taula.

Després de tractar les diferents taules de forma separada es procedeix a fusionar-les mitjançant la funció *Pandas.Dataframe.merge* en un únic *DataFrame* desat en la variable *dades_tot*.

Nomenclatura

Les assignatures estan representades inicialment per un codi amb el qual l'escola facilita la gestió. Visualment són difícils d'interpretar, motiu pel qual es substitueix el codi per un codi alfabètic.

Per tal de poder fer la substitució, s'ha creat una taula com la que es mostra on apareix cada assignatura amb el seu corresponent codi, nom i codi alfabètic emprat. A partir d'ella es du a terme la correspondència codi-codi alfabètic:

CODINUM	ASSIGNATURA	CODIALF
240011	Àlgebra Lineal	ALG
240012	Càlcul I	CALC1
240013	Mecànica Fonamental	MEC
240014	Química I	QUIM1
240015	Fonaments d'Informàtica	FONINFO
240021	Geometria	GEO
240022	Càlcul II	CALC2
240023	Termodinàmica Fonamental	TERMOFON
240072	Química II	QUIM2
240073	Expressió Gràfica	EXPRE
240171	Electromagnetisme	ELECTRO
240172	Mètodes Numèrics	METNUM
240180	Materials	MAT
240649	Equacions Diferencials	EQDIF
240626	Informàtica	INFO
240635	Mecànica	MEC

Taula 1. Correspondència d'assignatures i codis

Conversió de format

Pensant en els futurs algorismes de predicció que s'utilitzaran, existeix la possibilitat que aquests només acceptin una sèrie de formats de dades. Aquest és per exemple el cas

de l'arbre de decisió que s'utilitza i s'explicarà més endavant: només admet variables d'entrada numèriques.

En les dades que s'estan preparant, es tenen variables de format *string*. Una d'elles és la columna '*SEXE'*, on s'han reemplaçat els valors 'H' per 0 i els valors 'D' per 1 per tal que siguin numèrics.

Una altra variable que conté valors *string* és '*GRUP_CLASSE'*, en la qual les convocatòries que han sigut convalidades prenen el valor 'CONV'. Al comprovar que el nombre d'expedients on s'ha convalidat alguna assignatura és mínima respecte el nombre d'expedients totals, s'ha optat per descartar-los.

Enriquiment i millora

Fins ara tots els canvis realitzats en les dades han sigut a partir de les variables inicials. És possible que aquestes variables no siguin suficients, l'addició de més variables a partir de la informació que ja tenim permet tenir en compte més aspectes que d'alguna forma poden quedar amagades, o englobar en una única variable certa informació dispersa.

S'opta per la construcció de noves columnes/variables amb variables ja existents que poden resultar interessants en el posterior anàlisi de dades:

- *N_ASS_SUSP_FASEIN*
  Indica el nombre d'assignatures suspeses en la fase inicial per cada expedient a partir del càlcul del nombre de convocatòries cursades a cada assignatura. És a dir, indica el número d'assignatures que no s'han aprovat a la primera convocatòria i s'han hagut de tornar a matricular. Per exemple, en el cas d'un estudiant que ha aprovat totes les assignatures de fase inicial a la primera convocatòria, la variable *N_ASS_SUSP_FASEIN* pren valor 0.

- *N_QUAD_FASEIN*
  Indica el nombre de quadrimestres cursats durant la fase inicial a partir de la primera i última convocatòries cursades a la fase inicial. En el cas d'un estudiant que ha superat la fase inicial en 2 quadrimestres, la variable *N_QUAD_FASEIN* pren valor 2.

- *NMITJ_FASEIN*

  Indica la nota mitjana obtinguda en la fase inicial tenint en compte totes les convocatòries cursades, tant aprovades com suspeses.

- *GRUP_ACCES*

  Indica el primer grup en què cada expedient es va matricular durant tota l'estada en l'escola. Els grups s'han tractat de forma que acabin en 0, de forma que la variable pot prendre valors des de 10 fins a 100 de deu en deu (10,20,...,100).

  S'ha trobat el cas d'un petit grup d'expedients en què el primer grup d'accés és major a 100. S'ha comprovat que aquests casos corresponen a un mateix quadrimestre on es va donar un cas excepcional. Per tal de que aquesta minoria no influeixi en els resultats de la majoria, s'ha decidit descartar aquests expedients.

3. MODELATGE I VALIDACIÓ

Un cop s'han obtingut les dades preparades i estructurades en la forma desitjada, es procedeix al seu anàlisi. L'anàlisi comprèn dos fases: la construcció del model de predicció i la validació d'aquest. Cal dir que aquestes dues fases van agafades a la mà i es duen a terme alhora: la construcció del model implica la validació d'un model inicial i l'ajustament d'aquest a partir dels resultats obtinguts, es tracta d'un procés cíclic.

3.1. Classificació

L'anàlisi de dades a realitzar s'aplica en dades etiquetades per classes, és a dir, per atributs en què el valor a predir està basat en el valor d'altres atributs. En aquest cas, l'atribut a predir és la qualificació d'una assignatura de la fase no inicial del grau en funció d'altres atributs com les qualificacions en les assignatures corresponents a la fase inicial.

Dins d'aquest tipus d'anàlisi, es poden diferenciar dos grans classes:

- **Classificació**: Procés de cerca d'un model que descriu o distingeix classes de dades, el qual s'utilitza per predir el valor categòric que prendrà un element. Aplicat al projecte, un exemple seria la predicció de l'aprovat o suspès d'un estudiant en una assignatura determinada.
- **Regressió**: Mètode estadístic que també es basa en la predicció. En aquest cas, en comptes de predir valors categòrics, es prediuen valors numèrics. Recuperant l'exemple anterior, un exemple d'anàlisi regressiu seria la predicció de la nota quantitativa d'un estudiant en comptes d'una nota categòrica, on la nota pot prendre qualsevol valor nombre real comprès entre 0 i 10.

En aquest treball l'estudi s'enfocarà en mecanismes de classificació. Els classificadors, nom que reben els mètodes predictius de classificació, permeten un major enteniment de les dades i la millora o evitació de certes situacions futures. La detecció de frau en entitats financeres o la identificació de perfils de consumidors en comerços són exemples d'aplicació real d'aquest tipus de tècniques.

El modelatge del treball es centrarà en l'estudi de classificadors, la seva aplicació i la posterior validació. Els algoritmes a emprar s'extrauran de la llibreria sk-learn, de forma que es troben ja programats i preparats per poder aplicar-los directament.

Existeixen nombrosos algorismes de predicció basats en diferents metodologies. Algorismes com el *SVM* (*Support Vector Machine*) són de caire purament matemàtic, on les mostres són representades com punts de l'espai i l'algorisme va calculant subespais de classes, finalitzant quan els punts estan assignats a subespais els quals es troben a la distància màxima possible entre ells. Per altra banda es troben algorismes més visuals com els arbres de decisió, on l'algorisme crea una estructura de nodes que es divideixen en altres nodes aplicant una sèrie de condicions.

En el present treball s'ha seleccionat com a algorisme base l'arbre de decisió per dur a terme l'anàlisi. A més a més, aquest serà complementat per un algorisme d'acoblament anomenat *bagging* prenent com a estimador base l'arbre de decisió.

3.2. Arbres de decisió

Un arbre de decisió és un diagrama de flux estructurat en forma d'arbre que respon a una qüestió, on es parteix d'una arrel i mitjançant l'acompliment o no d'una sèrie de condicions es segueix un camí o altre que condueix finalment a una resposta. Els elements que el componen són:

- **Nodes** Conegudes també com a fulles, representen un atribut. Cada node es subdivideix en altres nodes a partir de l'aplicació de condicions.
    - **Node arrel** Primer atribut des d'on comença el diagrama i es desenvolupa en la tota la resta de nodes.
    - **Nodes terminals** Representen el final de l'arbre. Després de seguir un camí dins del diagrama, s'arriba al node terminal que indica la resposta o atribut corresponent a la qüestió plantejada.
    - **Nodes interns** Són tots els nodes que es troben entre el node arrel i els nodes terminals. Cada node intern es distribueix en més nodes, de forma que cada un d'ells és arrel d'un subarbre contingut dins l'arbre principal.
- **Branques** Consisteixen en unions entre nodes i condueixen a un node o altre depenent de la resposta a l'atribut del node d'on procedeix. Cada branca surt d'un únic node i condueix a un altre sol node.

L'explicació és més senzilla si es realitza mitjançant un exemple gràfic. A la *Figura 8* es troba un arbre de decisió que indica quines condicions defineixen la qualificació (aprovat

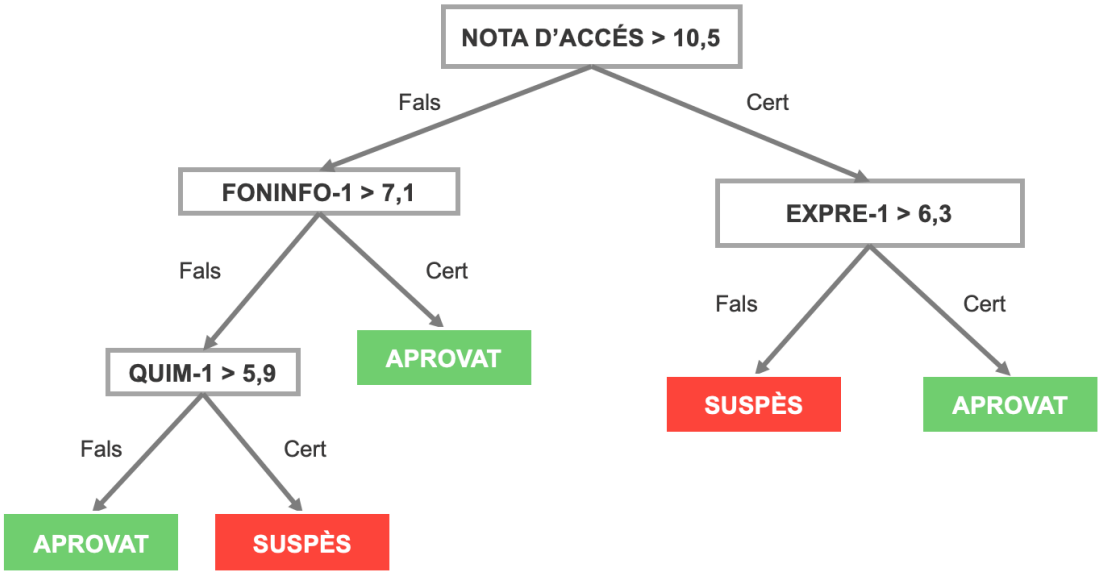

Figura 8. Predicció de la nota d'Informàtica. Exemple d'arbre de decisió.

o suspès) en l'assignatura *Informàtica* cursada per primer cop (primera convocatòria). L'arbre s'ha construït com a exemple, les respostes que inclou no son verídiques.

En l'exemple, els requadres corresponen a nodes: el primer de tot, amb l'atribut 'nota d'accés > 10,5', és el node arrel. A partir d'ell sorgeixen tots els nodes interns fins arribar als nodes acolorits que es tracten dels nodes terminals, indicant l'elecció final al camí seguit. Les fletxes entre requadres són les branques, indicant les possibles opcions a escollir a partir del node d'on surt (*Cert* si es compleix la condició o *Fals* en cas contrari). Aplicant l'arbre, si la nota d'accés de l'estudiant fos menor a 10.5 i la nota de primera convocatòria en l'assignatura de *Fonaments d'informàtica* fos major a 7.1, l'estudiant aprovaria *Informàtica* en la primera convocatòria.

El procediment que segueix un algorisme per tal de construir un arbre de decisió es basa en l'anàlisi de dades per subconjunts. Partint del conjunt total de dades, es realitzen talls de dades en subconjunts aplicant determinades condicions fins que totes les dades del subconjunt prenen el mateix valor en la categoria a predir. L'algorisme de l'arbre de decisió empra la recursivitat en l'algorisme: un subconjunt és analitzat i dividit en altres subconjunts fins que cada subconjunt és pur (la variable a predir té el mateix valor en totes les dades).

La qüestió és: com defineix l'algorisme les condicions que apareixen com a nodes per realitzar talls en els conjunts de dades? Per respondre a aquesta pregunta s'ha d'introduir la mesura *gini*, la forma més comuna de mesurar desigualtats. En aquest cas

quan es parla de desigualtat, es refereix a la diferència de valors en la classe a predir. Quan es realitza un tall a partir d'un node, es calcula la desigualtat i es decideix la següent condició a aplicar depenent de com canvia aquesta desigualtat abans i després del node.

En termes generals, *gini* és un paràmetre matemàtic que calcula la probabilitat de no trobar el valor correcte d'una classe després d'un node. Quan un subconjunt és pur, el paràmetre pren valor 0 ja que s'ha trobat el valor correcte en tots el subconjunt de dades. Quan s'assoleix un subconjunt pur, s'assoleix un node terminal.

Model d'arbre de decisió emprat

L'arbre de decisió classificador utilitzat és el *sklearn.tree.DecisionTreeClassifier* i es caracteritza per admetre només dades d'entrada en format numèric. Per aquest motiu en la preparació de dades s'han hagut de substituir els valors descrits en altres formats per valors numèrics. En el cas dels *missing values*, aquests s'han substituït per una nombre numèric elevat, concretament de 1000.

En l'algorisme de predicció *sklearn.tree.DecisionTreeClassifier* es poden destacar dos paràmetres possibles a definir. La definició de paràmetres permet millorar la fiabilitat del model predictor i reduir el possible *overfitting*, concepte que s'explicarà més endavant. Els paràmetres són:

- ***max_depth*** Indica la màxima profunditat de l'arbre. S'entén com a profunditat el nombre de nodes que es recorre en el camí més llarg de l'arbre abans d'arribar a un node terminal. En el cas de l'exemple representat en la *Figura* 7, la profunditat de l'arbre és igual a 3. Si no s'especifica cap valor a aquest paràmetre, l'arbre es desenvolupa a la màxima profunditat possible per defecte.

- ***min_samples_leaf*** Defineix el mínim nombre de mostres que ha de contenir un node per ser considerat com a tal. Si es defineix un mínim de *x* mostres, un node no pot ser dividit en un altre que només consti de *x* mostres. Aquest paràmetre es pot definir de dos formes:
    - Nombre enter (format *int*): si el valor introduït és un nombre enter *y*, el mínim nombre de mostres que ha de contenir un node és *y*.
    - Nombre decimal (format *float*): si el valor introduït és un nombre decimal, indica la fracció de mostres respecte el total que ha de contenir un node. Per exemple, si s'introdueix un valor *z*, el mínim nombre de mostres serà (*z**n) sent n el nombre total de mostres disponibles.

3.3. Mètodes combinats. *Bagging*.

A més d'utilitzar un algorisme que implementi el mètode d'arbre de decisió, també s'utilitzarà un algorisme basat en mètodes combinats, anomenats també mètodes d'*ensemble*.

Els mètodes combinats utilitzen múltiples algorismes de predicció per obtenir un rendiment superior al que s'obtindria per mitjà d'un únic algorisme estimador. El fet d'analitzar les dades a partir d'algorismes des de diferents 'punts de vista' i combinar-los tots en un únic algorisme fa que la precisió del model augmenti, ja que es tenen en compte més casos que poden no haver-se considerat aplicant només un sol algorisme. Els models combinats presenten el desavantatge de ser difícils d'analitzar: incorporen una gran quantitat de models predictius i, tot i que donen un millor resultat, és difícil de comprendre intuïtivament quins factors en concret contribueixen a la millor presa de decisions.

Aquests tipus de models ajuden també a evitar el fenomen d'*overfitting* o sobreajust, el qual es produeix quan el model de predicció està massa ajustat al conjunt de dades a partir el qual s'ha construït. S'obté una precisió molt elevada al aplicar el model a les dades utilitzades en la seva construcció, però un cop s'aplica a un altre conjunt de dades, la precisió difereix molt.

A la *Figura 9* es representa aquest fenomen aplicat a una regressió. En el cas d'un ajust correcte, la línia segueix el comportament dels punts de forma aproximada. En el cas d'*overfitting*, la línia recorre tots els punts d'un en un produint-se un sobreajust. El primer model podria aplicar-se a altres dades, mentre que el segon és aplicable només a les mateixes dades amb les que s'ha construït la regressió, els resultats no són realistes.

Les tècniques de combinació s'han anat desenvolupant sobretot durant els darrers anys, entre les quals destaquen les següents:

- ***Bagging*** Tècnica que redueix la variància d'un model a partir de la creació d'una sèrie d'algorismes amb subconjunts de dades i posteriorment combinant-les a partir de la millor o pitjor contribució al resultat.

- ***Boosting*** Mètode iteratiu que construeix el nou model mitjançant la construcció d'un model inicial utilitzant tot el conjunt de dades. Enfocant-se en les dades on no ha actuat bé el model inicial, es construeix un segon model i s'aplica en combinació

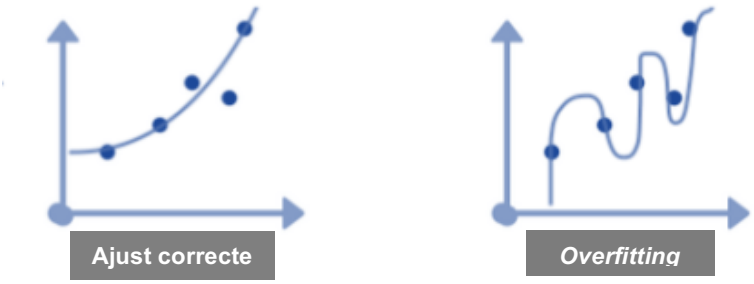

Figura 9. Ajust d'una regressió

amb el model inicial. Seguidament es construeix un tercer model tenint en compte les mancances de la combinació anterior i es realitza tot el procés repetidament, creant nous models fins obtenir-ne un amb una precisió acceptable.

- ***Stacking*** Combinació de diferents tipus de classificadors aplicant algorismes de meta-aprenentatge, en què s'estima quins mètodes són més fiables per naturalesa al problema analitzat.

Bagging

D'entre els diferents mètodes de combinació, en aquest projecte s'emprarà la tècnica *Bagging*. Aquest combina predictors d'un mateix estimador base, que en aquest cas serà l'arbre de decisió.

En la tècnica *Bagging*, el conjunt de dades per construir el model es distribueix en una sèrie de subconjunts de dades. La distribució es realitza de forma aleatòria i amb reemplaçament, és a dir, una dada determinada pot estar inclosa en més d'un subconjunt. A cada subconjunt de dades s'obté un algorisme predictor a partir de l'estimador base. La combinació de tots els predictors construïts mitjançant votació dóna lloc al predictor final: cada predictor prediu (vota) el valor categòric d'una instància en una classe determinada i el valor més votat de cada classe és el resultant en el predictor final. En la *Figura 10* es troba representat el procés de construcció descrit.

El model de mètode combinat *Bagging* emprat és *sklearn.ensemble.BaggingClassifier*, encarregat d'ajustar l'estimador base a cadascun dels subconjunts de dades creats aleatòriament, de forma que l'aleatorietat permet reduir la variància en la construcció del model. Els paràmetres a definir són els següents:

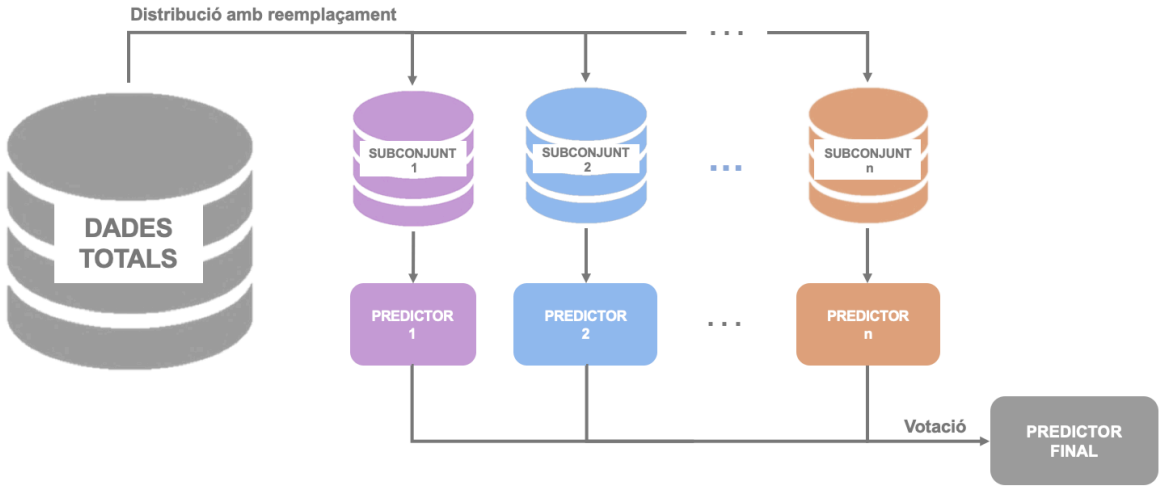

Figura 10. Construcció d'un model mitjançant Bagging

- ***base_estimator*** Estimador base a partir el qual es construeixen els predictors dels subconjunts de dades. En aquest cas es tractarà de l'arbre de decisió emprat (*sklearn.tree.DecisionTreeClassifier*).

- ***n_estimators*** Nombre de predictors que es creen a partir de l'estimador base. Si s'imposa un valor n, es crearan n classificadors que es combinaran per formar el classificador final.

- ***max_samples*** Màxim nombre de mostres que inclou cada subconjunt de dades, és a dir, el màxim nombre de mostres utilitzat en la construcció de cada classificador individual.

4. VALIDACIÓ

4.1. Mètodes de validació

Un cop construït el model, es necessita saber-ne la seva precisió. Tot i que es realitzen una sèrie d'ajustaments de model a partir d'un model inicial fins aconseguir una precisió vàlida, un procés convencional d'anàlisi de dades ha de passar una validació final del model predictiu. Com podem simular una situació real d'implementació del model?
Dos de les estratègies de validació més populars són les estratègies de *holdout* i k-*cross-validation*:

- **Holdout** Divisió de les dades en dos subconjunts: *training* i *testing*, on les de *training* s'utilitzen per construir el model i les de *testing* per validar-lo.
- **K-cross-validation** Divisió del conjunt de dades en k subconjunts on un es fa servir com a subconjunt *testing* i la resta de k-1 grups com a subconjunts *training*. Es tracta d'un mètode iteratiu on el procés es realitza k vegades prenent un subconjunt *testing* diferent cada cop.

En el projecte s'ha decidit utilitzar el mètode de validació *holdout.* Les dades es divideixen en dos conjunts, un de *training* i un de *testing,* que fan referència també a les dues fases de l'anàlisi de dades:

- ***Training*** La fase d'aprenentatge, coneguda com a *training*, consisteix en la construcció del classificador inicial a partir d'un conjunt de dades, anomenades mostres, aplicant un algoritme classificatori de predicció. Aquest classificador crearà unes normes a aplicar a totes les dades per decidir els valors que prenen en la variable a predir.

- ***Testing*** En la fase de t*esting* es valida el classificador inicial. Aquest classificador és aplicat a un conjunt de dades diferent a l'emprat en la seva construcció i es verifica la precisió del model.

Per tant, descrites les dues fases, les dades de *training* s'utilitzen per construir el model predictiu, mentre que les de *testing* es fan servir per avaluar la seva precisió i fiabilitat. En una aplicació correcta d'aquest tipus de validació, el model final es construeix a partir de les dades de *training* i les dades de *testing* s'haurien d'utilitzar només per una validació final.

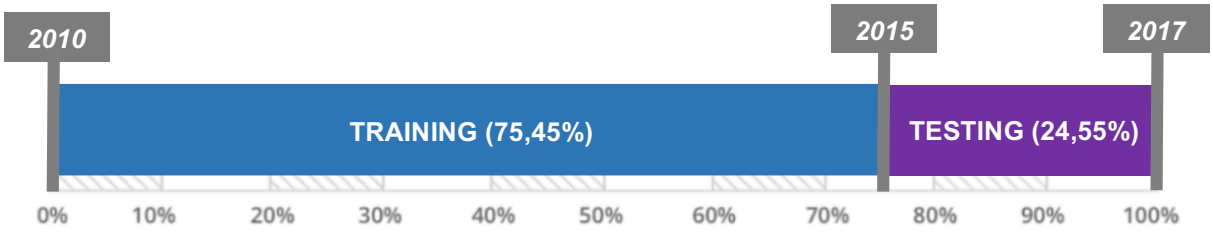

Figura 11. Partició del conjunt de dades

En el projecte la validació no s'ha pogut realitzar d'aquesta forma, donat que no es té un nombre suficient elevat de dades per poder realitzar aquest tipus de partició garantint que la construcció del model predictiu es realitza a partir d'un nombre de dades suficient.

Per tal de cobrir tant la construcció com la validació del model, és necessari dividir el total de dades disponibles en dos subconjunts: un de *training* i un de *testing*. Hi ha diferents formes de realitzar la partició, en aquest cas s'ha optat per separar les dades mitjançant un any de tall. L'any de tall imposat és 2015, de manera que a partir d'un total de dades que cobreixen el període 2010-2017, les dades de *training* corresponen al període a 2010-2014 i les dades de *testing* al període 2014-2017, tal i com mostra la *Figura 11*.

La possibilitat de dividir les dades mitjançant un any de tall és el motiu pel qual s'ha establert el *holdout* com a tècnica de validació. D'aquesta forma es té un major control de quines dades es fan servir i permet aplicar les mateixes dades en la construcció de models mitjançant diferents mètodes de predicció i poder comparar-los posteriorment.

S'ha editat una funció *traintest* que realitza la partició automàtica del conjunt total de dades. Introduint el conjunt de dades en format *DataFrame* i l'any de tall desitjat, retorna les dades de *training* i *testing* separades.

En general, la proporció de dades utilitzada pel *training* és d'un 70% respecte el total de i la resta, el 30%, s'utilitza per la part de *testing*. En el treball s'han aplicat uns percentatges aproximats: prenent com a any de tall 2015, s'obtenen 1801 mostres de *training* respecte un total de 2387 mostres, suposant un 75,45% de dades. La resta de dades, formada per 586 dades, es fan servir com a dades de *testing*.

Cal dir que la variable ANY_ACCES només s'utilitza per realitzar la divisió de dades, no s'inclou posteriorment en les variables emprades en la predicció. No té sentit incloure-la ja que l'any va augmentant a mesura que passa el temps, és a dir, no trobarem un

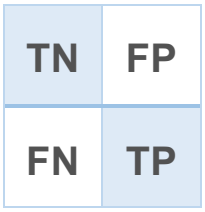

Figura 12. Matriu de confusió

any d'accés del 2012 per exemple intentant predir qualificacions d'estudiants amb any d'accés del 2016.

S'ha de realitzar també una partició entre les classes a partir les quals es fa la predicció i les classes que es desitgen predir. Les dades d'entrada de predicció les anomenarem variable X i s'introduiran com un *DataFrame*. La classe a predir, que serà la qualificació d'una assignatura determinada del Q3 serà la variable Y i es troba també en format de *DataFrame*.

De forma similar a la partició entre dades de *training* i *testing*, s'ha editat una funció *XY* per realitzar la partició automàtica entre dades de predicció X i dades a predir Y a partir de la definició de l'assignatura a predir.

4.2. Anàlisi de resultats

La predicció s'ha dut a terme per cadascuna de les 6 assignatures que composen el tercer quadrimestre del grau: *Electromagnetisme*, *Equacions diferencials*, *Informàtica*, *Materials*, *Mecànica* i *Mètodes numèrics*.

Primerament es construeix el model a partir de les dades de training i es calcula la precisió inicial. Posteriorment, a la validació es comprova si aquesta precisió es compleix comparant els valors de precisió aconseguits en el *training* i els obtinguts en el *testing*.

S'introdueix també la matriu de confusió com a eina per conèixer en quina part falla el model predictor. La matriu de confusió en aquest estudi té una dimensió de 2x2 al estar predient una classe binària, tal i com mostra la *Figura 12*. Els diferents elements de la matriu de confusió són:

- **TN** Nombre de dades que prenen valor 0 i han sigut predits correctament. Aplicat al treball, és el nombre de qualificacions que s'han predit com *Suspès* i es compleixen a la realitat.

- **FP** Nombre de dades que prenen valor 1 quan en realitat prenen valor 0. En el treball, és el nombre de qualificacions que s'han predit com *Aprovat* quan el seu valor real és *Suspès*.
- **FN** Nombre de dades que prenen valor 0 quan en realitat prenen valor 0. En el treball, és el nombre de qualificacions que s'han predit com *Suspès* quan el seu valor real és *Aprovat*.
- **TP** Nombre de dades que prenen valor 1 i han sigut predits correctament. Aplicat al treball, és el nombre de qualificacions que s'han predit com *Aprovat* i en la realitat també tenen valor *Aprovat*.

4.2.1. Predicció mitjançant arbres de decisions

L'anàlisi predictiu de dades s'ha realitzat primer aplicant algorismes basat en arbres de decisió. Tal i com s'ha descrit anteriorment, l'arbre de decisió emprat presenta dos paràmetres a definir, *max_depth* i *min_samples_leaf*. Recordem que per una banda *max_depth* és el nombre de nodes del camí més llarg a seguir dins l'arbre, mentre que per l'altra banda *min_samples_leaf* indica el mínim de mostres que ha de contenir un node.

La construcció d'arbres de decisió es fa aplicant diversos valors de *max_depth* i *min_samples_leaf*, on es calcula la precisió que atorga cadascun d'ells i s'avalua després quina és la precisió real que s'obté si s'apliquen en les dades de *testing*. El rang de valors pres pel paràmetre *max_depth* es troba entre 3 i 20, mentre que els valors de *min_samples_leaf* estan compresos entre 1 i 50.

Per poder estudiar les diferents precisions s'han construït taules on cada valor definit de *max_depth* és combinat amb cada valor definit de *min_samples_leaf* i viceversa. Les precisions estan expressades en tant per 1, de forma que un 0 indica la impossibilitat de predicció i un 1 correspon a un predictor ideal.

Predicció a l'assignatura *Electromagnetisme*

Les precisions dels diferents arbres de decisions en funció dels paràmetres *max_depth* i *min_samples_leaf* figuren en la *Taula 2*.

S'observa que en un valor fixe de *min_samples_leaf*, la precisió del model augmenta a mesura que va creixent el valor del paràmetre *max_depth*. Aquest fet era previsible, ja que a mesura que s'augmenta la profunditat de l'arbre, es dóna més opció a l'arbre d'ajustar-se a les dades amb les que es modela el mecanisme de predicció.

Si es fixa un valor de *max_depth*, la precisió del model tendeix a disminuir a mesura que va creixent el valor de *min_samples_leaf*. Aquest fet també era previsible, donat que per un valor major de *min_samples_leaf*, es limita més el lliure desenvolupament de l'arbre: al imposar un valor alt de mostres per node, impedeix la creació de nodes poc significatius i l'aparició del fenomen d'*overfitting*.

El valor ressaltat correspon a la combinació on el model assoleix la major precisió, la qual pren valor 1. 1 indica que el model està totalment ajustat a les dades amb les que s'ha construït (dades de *training*), fet que fa sospitar un possible *overfitting*. La combinació del model amb major precisió es dóna pel major *max_depth* i menor *min_samples_leaf*, la qual es tracta de la combinació que limita menys l'arbre.

Un cop construïts els diferents arbres de decisió, es valida la seva precisió sobre un conjunt de dades que encara no ha vist el model: les dades de *testing*. Les precisions obtingudes en cada arbre es troben a la *Taula 3.*

Simulant la situació real, s'observa que les precisions reals difereixen molt dels valors obtinguts en la *Taula 2*. L'arbre de decisió que havia obtingut la màxima precisió en el modelatge (1), un cop aplicat el mecanisme en unes altres dades ha obtingut la menor precisió (0,6092), suposant una disminució de 0,3908.

Els models que han obtingut major precisió després de ser aplicats a les dades de *training* són aquells que tenen un *min_samples_leaf* de 35 i prenen un valor de *max_depth* comprès entre 10 i 20. La precisió en el modelatge era de 0,8101 i aplicat a les noves dades ha passat a ser de 0,7065, el qual suposa una disminució de 0,1279.

Prenent com a model definitiu l'arbre de decisió que té un *min_samples_leaf* de 35 i *max_depth* de 10, es calcula la seva matriu de confusió, la qual està representada a la *Figura 13.*

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,7829	0,8123	0,8651	0,9356	0,9684	0,9822	0,9911	0,9967	**1,0000**
3	0,7829	0,8101	0,8551	0,9067	0,9250	0,9378	0,9423	0,9423	0,9428
5	0,7829	0,8079	0,8490	0,8884	0,9001	0,9062	0,9095	0,9106	0,9089
10	0,7829	0,8034	0,8323	0,8540	0,8606	0,8617	0,8617	0,8617	0,8606
15	0,7829	0,7990	0,8212	0,8451	0,8456	0,8456	0,8456	0,8456	0,8456
20	0,7829	0,7962	0,8184	0,8257	0,8257	0,8257	0,8257	0,8257	0,8257
25	0,7829	0,7962	0,8118	0,8195	0,8195	0,8195	0,8195	0,8195	0,8195
30	0,7829	0,7962	0,8090	0,8101	0,8101	0,8101	0,8101	0,8101	0,8101
35	0,7835	0,7968	0,8073	0,8101	0,8101	0,8101	0,8101	0,8101	0,8101
40	0,7873	0,8012	0,8012	0,8012	0,8012	0,8012	0,8012	0,8012	0,8012
50	0,7873	0,7996	0,7996	0,7996	0,7996	0,7996	0,7996	0,7996	0,7996

Taula 2. Precisions inicials de l'arbre de decisió en l'assignatura d'Electromagnetisme

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,6877	0,6826	0,6553	0,6416	0,6331	0,6263	0,6229	0,6143	0,6092
3	0,6877	0,6860	0,6638	0,6331	0,6314	0,6382	0,6348	0,6246	0,6451
5	0,6877	0,6928	0,6536	0,6382	0,6331	0,6433	0,6416	0,6416	0,6365
10	0,6877	0,6843	0,6706	0,6553	0,6433	0,6468	0,6416	0,6468	0,6433
15	0,6877	0,6877	0,6843	0,6809	0,6809	0,6809	0,6809	0,6809	0,6809
20	0,6877	0,6962	0,6809	0,6672	0,6672	0,6672	0,6672	0,6672	0,6672
25	0,6877	0,6894	0,6843	0,6860	0,6860	0,6911	0,6911	0,6911	0,6860
30	0,6877	0,6894	0,7048	0,6877	0,6877	0,6877	0,6877	0,6877	0,6877
35	0,6877	0,6775	0,6945	**0,7065**	**0,7065**	**0,7065**	**0,7065**	**0,7065**	**0,7065**
40	0,6843	0,6809	0,6809	0,6809	0,6809	0,6809	0,6809	0,6809	0,6809
50	0,6843	0,6655	0,6655	0,6843	0,6655	0,6843	0,6655	0,6843	0,6655

Taula 3. Precisions reals de l'arbre de decisió en l'assignatura d'Electromagnetisme

91	91
81	323

Figura 13. Matriu de confusió de l'arbre de decisió per Electromagnetisme

Els requadres en blau són el nombre de dades que s'han predit correctament: hi ha 91 suspesos i 323 aprovats ben predits. Els requadres en blau indiquen el nombre dades que s'han predit malament: 91 suspesos s'han predit com aprovats i 81 aprovats s'han predit com suspesos.

Veiem que hi ha una precisió de 0,5 en la predicció de suspesos i una precisió de 0,7995 79,95 en la predicció d'aprovats. Per tant, es pot dir que en la predicció de l'aprovat o suspès de l'assignatura d'*Electromagnetisme* mitjançant l'arbre de decisió com a model, es pot assolir una precisió de **0,7065** on la predicció de l'aprovat és molt més fiable que la del suspès.

Predicció a l'assignatura *Equacions diferencials*

Procedim a la construcció d'arbres de decisió per predir l'aprovat o suspès de l'assignatura *Equacions diferencials*, combinant diversos valors de paràmetres. Les precisions dels models es troben a la *Taula 4.*

S'observa que la precisió augmenta quan es limita menys la creació de l'arbre, és a dir, quan creix el paràmetre *max_depth* i quan disminueix *min_samples_leaf*. Per aquest motiu la major precisió del model, la qual és de 1, es dóna pel valor màxim de *max_depth* i el valor mínim de *min_samples_leaf*.

A continuació s'apliquen els arbres de decisió modelats a les dades de *testing*. Com es pot apreciar a la *Taula 5*, la major precisió s'obté en aquells models que en un principi tenien pitjor precisió.

En els models ressaltats, la precisió real és de 0,8737 mentre que el model en principi tenia una precisió de 0,8151. Això suposa un increment de precisió de 0,0671. Els models que tenien una precisió de 1 sobre el *training*, aplicat a noves dades passen a tenir una precisió de quasi un 0,3 menor.

Els models de major precisió després de ser aplicats a les dades de *training* són aquells que tenen un *max_depth* de 3 independentment del valor de *min_samples_leaf.* Aquest fet indica que per un *max_depth* igual a 3, la precisió es manté per molt que es variï el valor de *min_samples_leaf*, és a dir, els talls en les dades en busca de subconjunts purs no varien.

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,8151	0,8512	0,8934	0,9517	0,9789	0,9950	**1,0000**	**1,0000**	**1,0000**
3	0,8151	0,8501	0,8840	0,9334	0,9500	0,9511	0,9534	0,9534	0,9539
5	0,8151	0,8468	0,8717	0,9056	0,9156	0,9184	0,9178	0,9178	0,9189
10	0,8151	0,8418	0,8551	0,8712	0,8740	0,8723	0,8723	0,8740	0,8740
15	0,8151	0,8373	0,8523	0,8573	0,8579	0,8573	0,8573	0,8579	0,8579
20	0,8151	0,8345	0,8406	0,8473	0,8451	0,8473	0,8451	0,8451	0,8451
25	0,8151	0,8318	0,8395	0,8395	0,8395	0,8418	0,8418	0,8418	0,8395
30	0,8151	0,8284	0,8362	0,8362	0,8334	0,8362	0,8334	0,8334	0,8334
35	0,8151	0,8273	0,8323	0,8345	0,8345	0,8345	0,8345	0,8345	0,8345
40	0,8151	0,8262	0,8334	0,8334	0,8312	0,8334	0,8334	0,8334	0,8334
50	0,8151	0,8273	0,8279	0,8295	0,8279	0,8279	0,8279	0,8295	0,8279

Taula 4. Precisions inicials de l'arbre de decisió en l'assignatura d'Equacions diferencials

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	**0,8737**	0,8430	0,7901	0,7577	0,7440	0,7184	0,7270	0,7406	0,7218
3	**0,8737**	0,8447	0,7696	0,7116	0,7048	0,7167	0,7287	0,7116	0,7014
5	**0,8737**	0,8498	0,7969	0,7679	0,7611	0,7577	0,7389	0,7645	0,7594
10	**0,8737**	0,8396	0,7969	0,7355	0,7338	0,7372	0,7338	0,7423	0,7338
15	**0,8737**	0,8584	0,7816	0,7611	0,7560	0,7491	0,7491	0,7491	0,7611
20	**0,8737**	0,8413	0,8276	0,7952	0,7952	0,8140	0,8140	0,7833	0,8140
25	**0,8737**	0,8345	0,8106	0,8106	0,8106	0,8106	0,8191	0,8106	0,8191
30	**0,8737**	0,8328	0,8123	0,8123	0,8123	0,8157	0,8157	0,8157	0,8157
35	**0,8737**	0,8498	0,8294	0,8328	0,8328	0,8328	0,8294	0,8294	0,8294
40	**0,8737**	0,8532	0,8311	0,8276	0,8311	0,8328	0,8328	0,8311	0,8276
50	**0,8737**	0,8328	0,8584	0,8413	0,8584	0,8584	0,8584	0,8584	0,8584

Taula 5. Precisions reals de l'arbre de decisió en l'assignatura d'Equacions diferencials

En els models ressaltats, la precisió real és de 0,8737 mentre que el model en principi tenia una precisió de 0,8151. Això suposa un increment de precisió de 0,0671. Els models que tenien una precisió de 1 sobre el *training*, aplicat a noves dades passen a tenir una precisió de quasi un 0,3 menor.

27	38
36	485

Figura 14. Matriu de confusió de l'arbre de decisió per Equacions diferencials

Els models de major precisió després de ser aplicats a les dades de *training* són aquells que tenen d'un *max_depth* de 3 independentment del valor de *min_samples_leaf.* Aquest fet indica que per un *max_depth* igual a 3, la precisió es manté per molt que es variï el valor de *min_samples_leaf*, és a dir, els talls en les dades en busca de subconjunts purs no varien.

Es pren com a model definitiu l'arbre de decisió que té un *min_samples_leaf* de 1 i *max_depth* de 3. La seva matriu de confusió està representada a la *Figura 14.*

Mitjançant el model definitiu obtenim 27 suspesos i 485 aprovats ben predits. Per una altra banda, 38 suspesos s'han predit com aprovats i 36 aprovats s'han predit com suspesos. Hi ha una precisió de 0,4154 en la predicció de suspesos i una precisió de 0,9309 en la predicció d'aprovats.

Per tant, es pot dir que la predicció de l'aprovat o suspès de l'assignatura d'*Equacions diferencials*, mitjançant l'arbre de decisió com a model, pot assolir una precisió de **0,8737** on la predicció de l'aprovat és molt més precisa que la del suspès (el model prediu bé menys de la meitat dels suspesos).

Predicció a l'assignatura *Informàtica*

Continuem amb l'aplicació d'arbres de decisions en la predicció de l'aprovat en l'assignatura d'*Informàtica*. Les precisions dels models construïts a partir de les dades de *training* es mostren a la *Taula 6*.

Similarment a casos anteriors, la màxima precisió s'obté pel valor màxim de *max_depth* i valor mínim de *min_samples_leaf*. S'observa que les precisions són altes en tots els casos, sent totes majors a 0,834. A la validació es comprova si aquestes precisions s'aproximen a la realitat.

		max_depth								
		3	5	7	10	12	14	16	18	20
min_samples_leaf	1	0,8362	0,8623	0,8790	0,9206	0,9489	0,9667	0,9795	0,9911	**0,9978**
	3	0,8356	0,8612	0,8717	0,9062	0,9284	0,9367	0,9434	0,9522	0,9556
	5	0,8356	0,8579	0,8701	0,9023	0,9134	0,9184	0,9195	0,9217	0,9200
	10	0,8356	0,8501	0,8629	0,8717	0,8767	0,8795	0,8812	0,8812	0,8806
	15	0,8412	0,8501	0,8606	0,8606	0,8634	0,8634	0,8634	0,8634	0,8634
	20	0,8401	0,8462	0,8540	0,8540	0,8540	0,8540	0,8540	0,8540	0,8540
	25	0,8395	0,8440	0,8501	0,8512	0,8512	0,8512	0,8512	0,8512	0,8512
	30	0,8395	0,8418	0,8418	0,8462	0,8462	0,8462	0,8462	0,8462	0,8462
	35	0,8395	0,8406	0,8406	0,8451	0,8451	0,8451	0,8451	0,8451	0,8451
	40	0,8395	0,8406	0,8406	0,8451	0,8451	0,8451	0,8451	0,8451	0,8451
	50	0,8340	0,8401	0,8401	0,8401	0,8401	0,8401	0,8401	0,8401	0,8401

Taula 6. Precisions inicials de l'arbre de decisió en l'assignatura d'Informàtica

A la *Taula 7* es troben les precisions obtingudes un cop aplicats els models a les dades de *testing*. La situació és semblant al cas de predicció de l'assignatura *Equacions diferencials*: en la predicció sobre l'assignatura *Informàtica*, el model que tenia major precisió en principi (0,9978) ha passat a tenir una precisió de 0,7440, suposant una disminució de 0,2544.

Un cop més, la major precisió dels models aplicats en les noves dades s'obté per la profunditat mínima de l'arbre, amb un nombre de mostres per node comprès entre 25 i 40, on la precisió és de 0,8089. En la construcció dels models la precisió era de 0,8395, de manera que només ha disminuït un 0,0365.

Es pren com a model definitiu aquell que té un *min_samples_leaf* de 25 i un *max_depth* de 3. Es representa la matriu de confusió d'aquest model en la *Figura 15*.

S'observa que el nombre de dades predites correctament és de 35 en el cas de suspesos i 437 en el cas d'aprovats. Aquests equivalen a una precisió de 0,2612 en la predicció de suspesos i de 0,9668 en la d'aprovats.

Per tant, es considera que la predicció de l'aprovat o suspès en l'assignatura *Informàtica* es pot assolir amb una precisió del **0,8089** on la predicció de l'aprovat és molt més precisa que la del suspès. Amb una precisió tan baixa dels suspesos aquest model seria vàlid només per la precisió de l'aprovat.

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,8055	0,7850	0,7867	0,7765	0,7713	0,7526	0,7526	0,7440	0,7440
3	0,8055	0,7884	0,7884	0,7543	0,7594	0,7543	0,7509	0,7474	0,7321
5	0,8055	0,7884	0,7747	0,7850	0,7628	0,7782	0,7679	0,7577	0,7543
10	0,8055	0,7867	0,7730	0,7799	0,7628	0,7577	0,7509	0,7509	0,7423
15	0,8072	0,7816	0,7799	0,7645	0,7730	0,7713	0,7713	0,7730	0,7713
20	0,8072	0,7918	0,7867	0,7867	0,7867	0,7867	0,7867	0,7867	0,7867
25	**0,8089**	0,7867	0,7850	0,7765	0,7765	0,7765	0,7765	0,7765	0,7765
30	**0,8089**	0,7918	0,7918	0,7918	0,7918	0,7918	0,7918	0,7918	0,7918
35	**0,8089**	0,7952	0,7952	0,7952	0,7952	0,7952	0,7952	0,7952	0,7952
40	**0,8089**	0,7952	0,7952	0,7952	0,7952	0,7952	0,7952	0,7952	0,7952
50	0,8072	0,8055	0,8055	0,8055	0,8055	0,8055	0,8055	0,8055	0,8055

Taula 7. Precisions reals de l'arbre de decisió en l'assignatura d'Informàtica

Predicció a l'assignatura *Materials*

S'aplica el mètode d'arbre de decisió en la predicció de

35	99
15	437

Figura 15. Matriu de confusió de l'arbre de decisió per Informàtica

l'aprovat de l'assignatura *Materials*. Les precisions calculades dels diferents models en funció dels paràmetres *max_depth* i *min_samples_leaf* es troben en la *Taula 8*.

S'aprecia que la major precisió s'obté en els valors de paràmetres que limiten menys la construcció de l'arbre. La major precisió té valor 1, que pot haver-se donat segurament per un *overfitting* del model sobre les dades de *training*. La tendència de la precisió és similar als casos anteriors, creix a mesura que augmenta *max_depth* i disminueix quan s'augmenta *min_samples_leaf.*

Les precisions obtingudes després d'aplicar els models en les dades de *testing* es troben a la *Taula 9*. S'observa que les precisions són molt baixes respecte les prediccions en altres assignatures.

min_samples_leaf	max_depth 3	5	7	10	12	14	16	18	20
1	0,7735	0,8084	0,8534	0,9245	0,9606	0,9789	0,9928	0,9994	1,0000
3	0,7735	0,8073	0,8456	0,9073	0,9306	0,9406	0,9495	0,9495	0,9484
5	0,7735	0,8062	0,8440	0,8851	0,8973	0,9034	0,9062	0,9056	0,9073
10	0,7690	0,8023	0,8273	0,8523	0,8534	0,8529	0,8529	0,8529	0,8534
15	0,7690	0,8029	0,8207	0,8395	0,8395	0,8395	0,8395	0,8395	0,8395
20	0,7690	0,8007	0,8140	0,8223	0,8223	0,8223	0,8223	0,8223	0,8223
25	0,7690	0,7968	0,8073	0,8107	0,8107	0,8129	0,8107	0,8129	0,8107
30	0,7690	0,7946	0,8057	0,8040	0,8040	0,8057	0,8057	0,8040	0,8040
35	0,7690	0,7962	0,7962	0,7984	0,7984	0,7984	0,7984	0,7984	0,7984
40	0,7690	0,7957	0,7957	0,7957	0,7957	0,7957	0,7957	0,7957	0,7957
50	0,7690	0,7946	0,7946	0,7946	0,7946	0,7946	0,7946	0,7946	0,7946

Taula 8. Precisions inicials de l'arbre de decisió en l'assignatura Materials

min_samples_leaf	max_depth 3	5	7	10	12	14	16	18	20
1	0,6348	0,6621	0,6502	0,6399	0,6280	0,6246	0,6536	0,6468	0,6348
3	0,6348	0,6604	0,6331	0,6229	0,6109	0,6229	0,6212	0,6126	0,6143
5	0,6348	0,6689	0,6280	0,6314	0,6177	0,6331	0,6229	0,6195	0,6399
10	0,6382	0,6724	0,6297	0,6416	0,6433	0,6468	0,6468	0,6451	0,6451
15	0,6382	0,6621	0,6109	0,6229	0,6229	0,6229	0,6229	0,6229	0,6229
20	0,6382	0,6553	0,6007	0,6092	0,6109	0,6109	0,6092	0,6092	0,6092
25	0,6382	0,6672	0,6263	0,6212	0,6212	0,6212	0,6212	0,6212	0,6212
30	0,6382	0,6587	0,6519	0,6553	0,6519	0,6553	0,6519	0,6553	0,6553
35	0,6382	0,6741	0,6741	0,6655	0,6655	0,6655	0,6655	0,6655	0,6655
40	0,6382	0,6689	0,6689	0,6689	0,6689	0,6689	0,6689	0,6689	0,6689
50	0,6382	0,6621	0,6621	0,6621	0,6621	0,6621	0,6621	0,6621	0,6621

Taula 9. Precisions reals de l'arbre de decisió en l'assignatura Materials

Tot i que la major precisió no es dóna al valor mínim de *max_depth*, sí que l'obtenim a valors baixos. Concretament, la precisió màxima és de 0,6741. Aquest model aplicat a les dades de *training* tenia una precisió del 0,7962, de forma que la seva precisió ha disminuït un 0,1534.

124	136
79	247

Figura 16. Matriu de confusió de l'arbre de decisió per Materials

Es pren com a model definitiu el que té una profunditat de 5 nodes on cada node conté un mínim de 35 mostres. La matriu de confusió, representada en la *Figura 16*, mostra que hi ha un una precisió de 0,4769 en la predicció del suspès i de 0,7577 en la predicció de l'aprovat.

Es pot concloure que l'aplicació de l'arbre de decisió en l'assignatura *Materials* ha ofert una precisió d'un **0,6741,** on la predicció de l'aprovat segueix sent molt més precisa que la del suspès, com en el cas d'altres assignatures.

Predicció a l'assignatura *Mecànica*

Es procedeix a aplicar l'arbre de decisió com a mètode de predicció per predir l'aprovat en l'assignatura de *Mecànica*. Les precisions obtingudes a cada model es troben a la *Taula 10*. Es torna a donar la màxima precisió, amb valor 1, al màxim valor de *max_depth* i mínim valor de *min_samples_leaf*.

En comparació amb les precisions inicials obtingudes en prediccions d'altres assignatures, les precisions de la taula són inferiors. Aquest resultat fa pensar ja que les precisions dels models aplicats a noves dades també seran inferiors.

Es procedeix a aplicar els arbres de decisió obtinguts sobre les dades de *testing*. Els resultats obtinguts en forma de precisions estan expressats en la *Taula 11*. Tal i com s'havia sospitat, la precisió màxima és baixa: té un valor de 0,6843.

El model on s'havia obtingut una precisió inicial de 1 ara passa a tenir una precisió de 0,5802, suposant una disminució del 0,4198. La diferència de precisions és molt gran, fet que s'atribueix a un *overfitting* del model a les dades de *training*.

La màxima precisió correspon a un model amb una profunditat de 7 nodes amb un mínim de 35 mostres per node. La precisió inicial del mateix model era de 0,7812, de forma

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,7313	0,7801	0,8490	0,9317	0,9650	0,9878	0,9961	0,9978	1,0000
3	0,7313	0,7785	0,8401	0,9023	0,9228	0,9400	0,9417	0,9439	0,9428
5	0,7313	0,7762	0,8284	0,8778	0,8873	0,8923	0,8934	0,8934	0,8934
10	0,7301	0,7712	0,8107	0,8345	0,8368	0,8368	0,8368	0,8368	0,8368
15	0,7301	0,7646	0,7957	0,8140	0,8140	0,8140	0,8140	0,8140	0,8140
20	0,7301	0,7612	0,7912	0,8023	0,8023	0,8023	0,8023	0,8023	0,8023
25	0,7301	0,7590	0,7812	0,7923	0,7923	0,7923	0,7923	0,7923	0,7923
30	0,7301	0,7574	0,7835	0,7879	0,7879	0,7879	0,7879	0,7879	0,7879
35	0,7301	0,7601	0,7812	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857
40	0,7301	0,7601	0,7801	0,7823	0,7823	0,7823	0,7823	0,7823	0,7823
50	0,7301	0,7590	0,7712	0,7712	0,7712	0,7712	0,7712	0,7712	0,7712

Taula 10. Precisions inicials de l'arbre de decisió en l'assignatura Mecànica

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,6331	0,6451	0,6331	0,5870	0,6024	0,5939	0,5922	0,6126	0,5802
3	0,6331	0,6451	0,6382	0,6246	0,6297	0,6331	0,6382	0,6280	0,6485
5	0,6331	0,6519	0,6451	0,6195	0,6280	0,6348	0,6433	0,6399	0,6331
10	0,6314	0,6416	0,6519	0,6348	0,6212	0,6195	0,6314	0,6177	0,6177
15	0,6314	0,6416	0,6570	0,6416	0,6416	0,6416	0,6416	0,6416	0,6416
20	0,6314	0,6382	0,6604	0,6382	0,6382	0,6382	0,6382	0,6382	0,6382
25	0,6314	0,6331	0,6519	0,6365	0,6365	0,6365	0,6365	0,6365	0,6365
30	0,6314	0,6416	0,6741	0,6536	0,6536	0,6536	0,6536	0,6536	0,6536
35	0,6314	0,6365	0,6843	0,6638	0,6638	0,6638	0,6638	0,6638	0,6638
40	0,6314	0,6365	0,6706	0,6502	0,6502	0,6502	0,6502	0,6502	0,6502
50	0,6314	0,6365	0,6485	0,6485	0,6485	0,6485	0,6485	0,6485	0,6485

Taula 11. Precisions reals de l'arbre de decisió en l'assignatura Mecànica

que la precisió del model aplicat a noves dades ha disminuït 0,124. Prenent aquest model com a model final, es calcula la matriu de confusió representada en la *Figura 17*.

En la matriu s'aprecia que un 0,7273 del total de suspesos estan predits correctament i un 0,5549 del total d'aprovats estan predits correctament. És la primera assignatura de les predites fins ara on la precisió de predicció és menor en l'aprovat que en el suspès.

168	63
158	197

Figura 17. Matriu de confusió de l'arbre de decisió per Mecànica

De fet, en la majoria d'assignatures el model fallava molt més en la predicció del suspès que en el de l'aprovat. La millor predicció del suspès es dóna perquè el nombre de suspesos (326) en *Mecànica* és major al nombre d'aprovats (260), permetent tenir més dades de suspesos per modelar el predictor.

Analitzats els resultats obtinguts, es pot afirmar que dins la precisió que ofereix el millor model, de **0,6843**, la fiabilitat de la predicció del suspès és major a la que proporciona per predir l'aprovat.

Predicció a l'assignatura *Mètodes numèrics*

S'apliquen arbres de decisions en la predicció de l'aprovat de l'última assignatura restant: *Mètodes numèrics*. Les precisions obtingudes per cada profunditat d'arbre combinada amb cada nombre de mostres per node es mostren a la *Taula 12*.

Com a primera observació, s'aprecia que les precisions tenen un valor molt elevat, sent major a 0,9 en tots els casos. Es tornen a donar casos d'*overfitting* en aquells arbres de precisió molt elevada on els valors de paràmetres són poc restrictius.

Els mateixos arbres de decisió creats en la construcció de models s'empren per la predicció de noves dades. Els resultats obtinguts es troben representats a la *Taula 13.* Es pren com a model definitiu aquell que té una profunditat de 3 nodes on cada node conté un mínim de 25 mostres, el qual té una precisió de 0,7952.

La matriu de confusió del model definitiu es troba representada en la *Figura 18*. La matriu mostra que hi ha una precisió de 0,1926 en la predicció del suspès i una precisió de 0,9290 en la predicció de l'aprovat. L'alt error en la predicció del suspès indica que el model no és fiable en la predicció de suspesos.

S'observa que en les dades de *testing*, el nombre d'aprovats és de 528 i el nombre de suspesos és de 58. Aquesta proporció és similar també en les dades de *training*, de

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,9200	0,9317	0,9539	0,9811	0,9906	0,9956	0,9994	**1,0000**	**1,0000**
3	0,9200	0,9300	0,9467	0,9667	0,9695	0,9722	0,9722	0,9717	0,9711
5	0,9200	0,9289	0,9400	0,9500	0,9528	0,9539	0,9528	0,9534	0,9506
10	0,9200	0,9245	0,9311	0,9339	0,9339	0,9339	0,9339	0,9339	0,9339
15	0,9184	0,9223	0,9250	0,9262	0,9262	0,9262	0,9262	0,9262	0,9262
20	0,9178	0,9178	0,9178	0,9178	0,9178	0,9178	0,9178	0,9178	0,9178
25	0,9150	0,9150	0,9150	0,9150	0,9150	0,9150	0,9150	0,9150	0,9150
30	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128
35	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128	0,9128
40	0,9112	0,9112	0,9112	0,9112	0,9112	0,9112	0,9112	0,9112	0,9112
50	0,9095	0,9095	0,9095	0,9095	0,9095	0,9095	0,9095	0,9095	0,9095

Taula 12. Precisions inicials de l'arbre de decisió en l'assignatura Mètodes numèrics

min_samples_leaf \ max_depth	3	5	7	10	12	14	16	18	20
1	0,7799	0,7730	0,7833	0,7628	0,7696	0,7611	0,7611	0,7679	0,7594
3	0,7799	0,7747	0,7782	0,7679	0,7645	0,7577	0,7628	0,7645	0,7543
5	0,7799	0,7730	0,7594	0,7645	0,7628	0,7628	0,7628	0,7611	0,7679
10	0,7799	0,7730	0,7713	0,7765	0,7765	0,7765	0,7765	0,7765	0,7765
15	0,7799	0,7816	0,7901	0,7901	0,7833	0,7833	0,7901	0,7833	0,7901
20	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935
25	**0,7952**	**0,7952**	**0,7952**	**0,7952**	**0,7952**	**0,7952**	**0,7952**	**0,7952**	**0,7952**
30	0,7918	0,7918	0,7850	0,7850	0,7850	0,7850	0,7850	0,7850	0,7850
35	0,7918	0,7918	0,7918	0,7918	0,7918	0,7918	0,7918	0,7918	0,7918
40	0,7850	0,7850	0,7850	0,7850	0,7850	0,7850	0,7850	0,7850	0,7850
50	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935	0,7935

Taula 13. Precisions reals de l'arbre de decisió en l'assignatura Mètodes numèrics

26	109
32	419

Figura 18. Matriu de confusió de l'arbre de decisió per Mètodes numèrics

forma que el model ha tingut moltes més dades d'aprovats que de suspesos per modelar quins patrons segueixen, fet que justifica la major precisió en la predicció d'aprovats.

Es pot concloure que l'aplicació de l'arbre de decisió en l'assignatura *Mètodes numèrics* ha ofert una precisió d'un **0,7952**, però el model és molt més fiable en la predicció d'aprovats.

Abans de tancar l'anàlisi dels arbres de decisió, es vol afegir un comentari sobre un fet que es produeix en moltes prediccions. En el cas de la predicció a l'assignatura de *Mètodes numèrics,* s'observa a la *Taula 13* que la major precisió es produeix en el conjunt d'arbres que tenen un mínim de 25 mostres per node, independentment de la profunditat de l'arbre. Aquest comportament es repeteix en la predicció d'altres assignatures: la precisió és la mateixa per un valor fixe de *min_samples_leaf* sense dependre de la profunditat de l'arbre. Aquest fenomen es deu a que per molts talls que realitzi l'arbre, el valor de la classe que prediu és el mateix. A la *Figura 16* es troben representats els arbres de decisió corresponents a una profunditat màxima de 3 i una profunditat màxima de 20, ambdós amb un mínim nombre de 25 mostres per node.

A la *Figura 19* es pot veure que a l'arbre de més profunditat, en els nodes que van més enllà que l'arbre de menys profunditat, el valor de la classe que prediu és el mateix en tots els nodes. En els nodes marcats en un mateix color, el valor predit és el mateix en tots, de manera que el desenvolupament de l'arbre es podria haver aturat en el primer node de mateix color. El procés de tall de tots aquells nodes que procedeixen d'un mateix node i tenen tots el mateix valor de classe es coneix com a **poda d'arbre**.

En el propi algorisme emprat al treball no s'incorpora la poda d'arbre, però existeixen diferents mecanismes de poda, els quals permeten obviar aquells nodes no significatius un cop construït l'arbre de decisió.

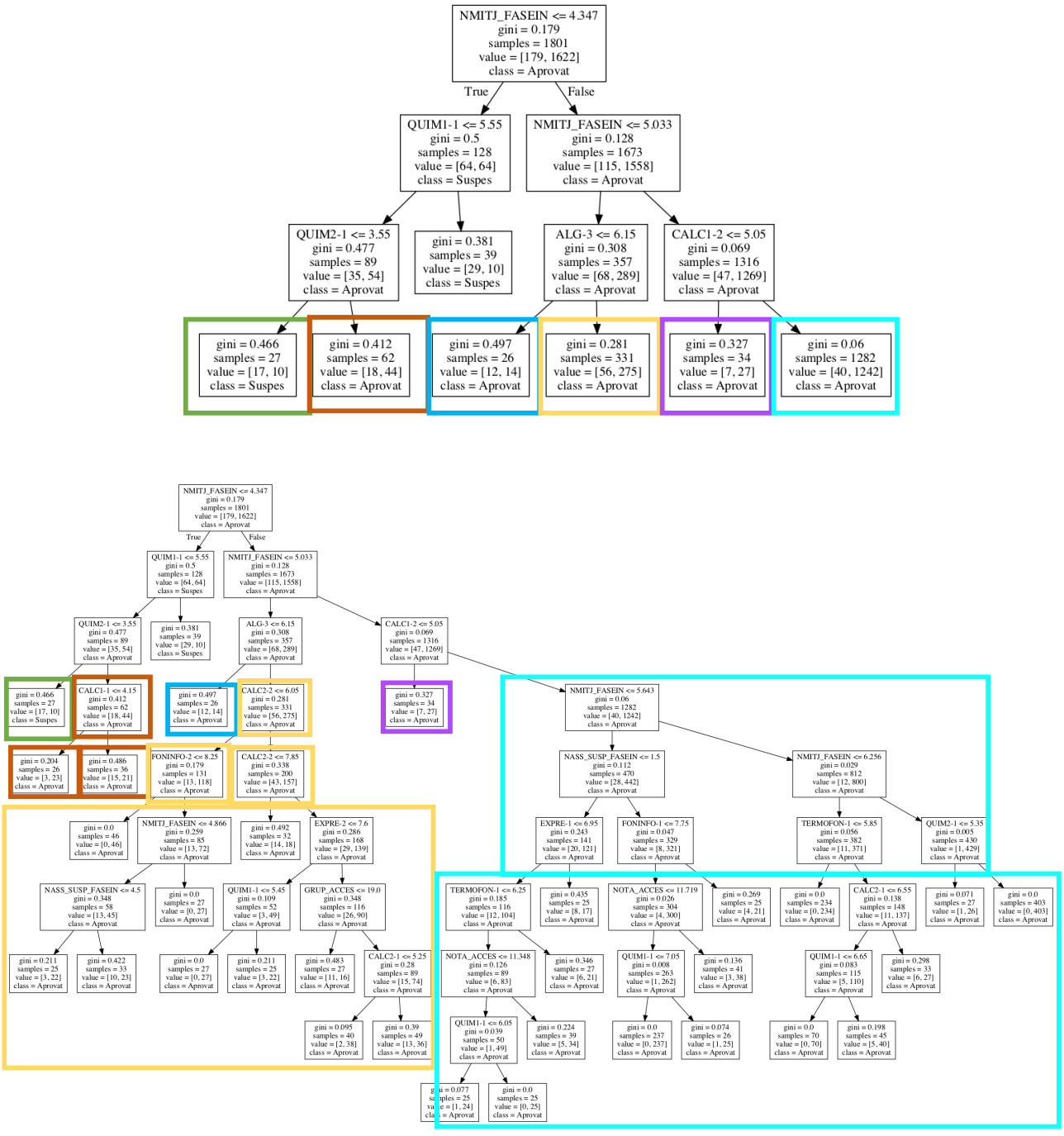

Figura 19. Comparació entre arbres de mateix min_samples_leaf i diferent max_depth

4.2.2. Predicció mitjançant *Bagging*

Assignatura	*max_depth*	*min_samples_leaf*
Electromagnetisme	10	35
Equacions diferencials	3	50
Informàtica	3	25
Materials	5	35
Mecànica	7	35
Mètodes numèrics	3	25

Taula 14. Paràmetres emprats en l'arbre de decisió com a estimador base

Després de realitzar les prediccions mitjançant arbres de decisió, es procedeix a aplicar el mateix procediment a partir del mètode combinat *Bagging*. Concretament, s'utilitzarà com a estimador base del *Bagging* els models definitius d'arbre de decisió trobats en les seccions anteriors. Els paràmetres *max_depth* i *min_samples_leaf* de l'estimador base variaran depenent de l'assignatura a predir. S'han pres els paràmetres dels models que millor prediuen les qualificacions de cada assignatura:

Recordem que l'algorisme *Bagging* presenta dos paràmetres a definir, *n_estimators* i *max_samples,* on *n_estimators* és el nombre d'estimadors emprats en la construcció del model final i *max_samples* el nombre màxim de mostres utilitzat en la construcció de cada estimador. En el cas de *n_estimators*, s'ha pres un rang de valors comprès entre 1 i 100, mentre que en el cas de *max_samples* els valors presos es troben entre 10 i 1500.

D'igual forma que amb els arbres de decisions, per l'estudi de resultats s'han construït taules on cada valor definit de *n_estimators* és combinat amb cada valor definit de *max_samples* i viceversa. Les precisions dels models estan expressades també en tant per 1, de forma que un 0 correspon a un predictor que no prediu cap valor correcte i un 1 correspon a un predictor ideal.

Predicció a l'assignatura *Electromagnetisme*

S'aplica la tècnica Bagging en la predicció de l'aprovat de l'assignatura *Electromagnetisme*. Les precisions obtingudes en funció dels paràmetres *n_estimators* i *max_samples* es representen a la *Taula 15*.

Es pot observar que, en general, la precisió augmenta a mesura que creix el valor de *max_samples*: quant major és el nombre de mostres que prenen els estimadors per construir cada predictor individual, la precisió del model final augmenta. En el cas del paràmetre *n_estimators*, la tendència no és tan clara. En el rang de valors comprès entre 1 i 10, la precisió augmenta amb *n_estimators*: quant major és el nombre d'estimadors individuals emprats per construir el predictor final, major precisió s'obté. A partir del valor de *n_estimators* superior a 10, la precisió no té un comportament clar.

La màxima precisió, de valor 0,8173, s'obté pel valor màxim definit de *max_samples* (1500) i per un nombre elevat d'estimadors (25). Comparant aquest valor amb l'obtingut després d'aplicar el model a les dades de *testing* (representat a la *Taula 16*), el qual és de 0,7065 segueix havent una disminució de precisió destacable de 0,1108.

Analitzant la *Taula 16*, corresponent a les precisions obtingudes al aplicar els models a les noves dades, observem que el model que té màxima precisió (0,7184) té uns valors paramètrics (*max*_samples=1000, *n*_estimators=50) similars als que s'havien obtingut en la construcció del model (*max*_samples=1500, *n*_estimators=25).

Prenem com a model definitiu aquell que utilitza 50 estimadors i 1000 mostres per estimador en la construcció del model final. La seva matriu de confusió es troba representada en la *Figura 20*.

Es torna a repetir el comportament que s'havia observat en les prediccions mitjançant arbres de decisió: la precisió en la predicció d'aprovats és molt major a la de suspesos. En aquest cas, hi ha una precisió de 0,1429 en la predicció de suspesos i un 0,9410 en la d'aprovats, una diferència de precisions encara major a l'arbre de decisió.

Per tant, es pot concloure que en la predicció mitjançant *Bagging* de l'aprovat o suspès en l'assignatura *Electromagnetisme*, el model pot assolir una precisió de **0,7184** sent molt més fiable en la predicció d'aprovats.

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830
20	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830
50	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830	0,6830
100	0,7346	0,7646	0,7535	0,7607	0,7696	0,7679	0,7707	0,7685	0,7690
150	0,7773	0,7696	0,7729	0,7768	0,7773	0,7757	0,7723	0,7740	0,7779
200	0,7768	0,7762	0,7718	0,7818	0,7829	0,7790	0,7762	0,7773	0,7779
250	0,7346	0,7690	0,7751	0,7790	0,7785	0,7740	0,7829	0,7812	0,7790
300	0,7385	0,7807	0,7674	0,7785	0,7862	0,7812	0,7801	0,7790	0,7796
500	0,7690	0,7946	0,7851	0,7957	0,7907	0,7896	0,7890	0,7907	0,7912
1000	0,7807	0,7951	0,8034	0,8001	0,8018	0,8090	0,8029	0,8040	0,8012
1500	0,7890	0,8029	0,8040	0,8118	0,8140	**0,8173**	0,8101	0,8118	0,8096

Taula 15. Precisions inicials del Bagging en l'assignatura Electromagnetisme

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894
20	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894
50	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894	0,6894
100	0,6877	0,6672	0,7014	0,7014	0,6877	0,6894	0,6860	0,6911	0,6894
150	0,6536	0,6860	0,6860	0,7014	0,6980	0,7014	0,7014	0,7099	0,7014
200	0,6007	0,6826	0,6877	0,6962	0,7031	0,6980	0,7048	0,7099	0,7065
250	0,6672	0,6980	0,7048	0,7082	0,7014	0,7082	0,7065	0,7065	0,7065
300	0,6894	0,6792	0,6928	0,7031	0,7065	0,7116	0,7116	0,7065	0,7133
500	0,6928	0,6826	0,7082	0,6997	0,7031	0,7099	0,7082	0,7031	0,7048
1000	0,7031	0,7184	0,7082	0,7116	0,7065	0,7099	**0,7184**	0,7150	0,7184
1500	0,6775	0,6962	0,7065	0,7065	0,7167	0,7065	0,7065	0,7099	0,7218

Taula 16. Precisions reals del Bagging en l'assignatura Electromagnetisme

26	156
24	380

Figura 20. Matriu de confusió del Bagging per Electromagnetisme

Predicció a l'assignatura *Equacions diferencials*

Es procedeix a predir l'aprovat o suspès en l'assignatura *Equacions diferencials*. Les precisions dels diferents models construïts es troben a la *Taula 17.* Igual que en l'assignatura d'*Electromagnetisme*, les precisions inicials són majors quan el nombre de mostres que pren cada estimador individual és major.

A la *Taula 18* es troben les precisions dels diferents models aplicats sobre les noves dades. A diferència del cas d'*Electromagnetisme*, en *Equacions diferencials* els models que tenen major precisió després de la validació són els que prenen menys dades per estimador individual.

Es pren com a model definitiu aquell que pren 3 estimadors i 10 mostres per estimador per la construcció del predictor final. Una característica a destacar és que la seva precisió al ser aplicat a les dades de *testing* és de 0,8891 mentre que aplicat a les dades de *training* és de 0,8240. És el primer model que es construeix que ha obtingut una major precisió sobre el *testing* que sobre el *training*, concretament un 0,0651 més. Aquest és el comportament que es desitja per cada predictor: que obtingui una major precisió en la seva aplicació sobre un cas real.

A la *Figura 21* es troba representada la matriu de confusió del model *Bagging* definitiu. El model té una precisió de 0,3846 predient el suspens i una precisió de 0,9520 predient l'aprovat. Tot i que la precisió en la predicció del suspens és major que en el cas de l'assignatura *Electromagnetisme*, aquesta segueix sent molt baixa.

Després d'analitzar els resultats, es pot concloure que el model *Bagging* aplicat a l'assignatura *Equacions diferencials* és un bon predictor, amb una precisió de **0,8891** tot i que la predicció del suspens sigui molt dolenta.

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857
20	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857
50	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857
100	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857
150	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857
200	0,7268	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857	0,7857
250	0,7857	0,7857	0,7829	0,7857	0,7812	0,7857	0,7857	0,7857	0,7857
300	0,7840	0,7701	0,7823	0,7885	0,7890	0,8034	0,7984	0,8046	0,7901
500	0,8123	0,7962	0,8090	0,8107	0,8157	0,8162	0,8118	0,8101	0,8107
1000	0,7984	0,8179	0,8190	0,8212	0,8195	0,8223	0,8218	0,8173	0,8179
1500	0,8140	**0,8240**	0,8223	**0,8240**	0,8218	0,8218	0,8207	0,8195	0,8234

Taula 17. Precisions inicials del Bagging en l'assignatura Equacions diferencials

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**
20	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**
50	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**
100	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**
150	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**	**0,8891**
200	**0,8891**	**0,8891**	**0,8891**	**0,8891**	0,8823	**0,8891**	**0,8891**	**0,8891**	**0,8891**
250	**0,8891**	**0,8891**	0,7833	**0,8891**	0,8891	0,8788	**0,8891**	**0,8891**	**0,8891**
300	0,7833	0,8584	0,8720	**0,8891**	0,8345	0,8840	0,8857	**0,8891**	0,8874
500	0,8584	0,8754	0,8549	0,8567	0,8635	0,8652	0,8618	0,8601	0,8635
1000	0,7918	0,8737	0,8720	0,8703	0,8618	0,8686	0,8720	0,8754	0,8754
1500	0,8362	0,8549	0,8703	0,8686	0,8788	0,8686	0,8703	0,8703	0,8788

Taula 18. Precisions reals del Bagging en l'assignatura Equacions diferencials

25	40
25	496

Figura 21. Matriu de confusió del Bagging per Equacions diferencials

Predicció a l'assignatura *Informàtica*

Continuem amb l'aplicació del *Bagging* en la predicció de l'aprovat en l'assignatura d'*Informàtica*. Les precisions dels models construïts a partir de les dades de *training* es mostren a la *Taula 19*.

La màxima precisió, amb un valor de 0,8373, s'obté per un model que utilitza 75 estimadors individuals, un nombre major al que s'utilitzava en les assignatures anteriorment predites. La precisió màxima obtinguda en els models aplicats en les dades de *testing* és de 0,8345, tal i com es veu a la *Taula 20*, i correspon a un model que utilitza també 75 estimadors individuals. El mateix model havia obtingut en el *training* una precisió de 0,8318, de forma que el model ha sigut 0,027 més precís en la validació.

Prenem com a model definitiu el *bagging* amb valor 75 en el paràmetre *n_estimators* i 250 en el paràmetre *max_samples*. La seva matriu de confusió està representada a la *Figura 22*. Per una banda, 35 suspesos s'han predit correctament mentre que hi ha 99 aprovats predits com suspesos. Per altra banda, 437 aprovats s'han predit correctament mentre que 11 suspesos s'han predit com aprovats. Aquests valors es tradueixen en una precisió de la predicció del suspès de 0,2612 i en el cas de l'aprovat, de 0,9757.

Es pot afirmar que el model *Bagging* aplicat a l'assignatura d'*Informàtica* pot assolir una precisió de **0,8345** en la predicció de l'aprovat o suspès, sent fiable només la predicció de l'aprovat.

		n_estimators								
		1	3	5	10	15	25	50	75	100
max_samples	10	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034
	20	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034
	50	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034
	100	0,8034	0,7951	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034	0,8034
	150	0,8218	0,8034	0,7946	0,8145	0,8046	0,8051	0,8046	0,8245	0,8218
	200	0,7851	0,8162	0,8273	0,8234	0,8240	0,8284	0,8262	0,8295	0,8279
	250	0,7896	0,8079	0,8240	0,8334	0,8290	0,8351	0,8329	0,8318	0,8262
	300	0,8084	0,8201	0,8290	0,8329	0,8290	0,8318	0,8356	0,8356	0,8334
	500	0,8329	0,8340	0,8373	0,8334	0,8345	0,8351	0,8356	0,8340	0,8356
	1000	0,8229	0,8362	0,8334	0,8368	0,8368	0,8362	0,8368	**0,8373**	0,8368
	1500	0,8090	0,8401	0,8401	0,8418	0,8390	0,8384	0,8351	0,8368	0,8384

Taula 19. Precisions inicials del Bagging en l'assignatura Informàtica

		n_estimators								
		1	3	5	10	15	25	50	75	100
max_samples	10	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,77´3	0,7713	0,7713
	20	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,77´3	0,7713	0,7713
	50	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713
	100	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713	0,7713
	150	0,8038	0,7577	0,7969	0,7901	0,8020	0,7935	0,7884	0,7850	0,7730
	200	0,7935	0,7713	0,8140	0,8072	0,8328	0,8225	0,8259	0,8208	0,8225
	250	0,7628	0,7935	0,8242	0,8259	0,8225	0,8174	0,8276	**0,8345**	0,8328
	300	0,7696	0,8259	0,8055	0,7986	0,8225	0,8294	0,8276	0,8259	0,8259
	500	0,7901	0,7901	0,8174	0,8174	0,8225	0,8208	0,8191	0,8191	0,8174
	1000	0,7628	0,8089	0,8072	0,8106	0,8123	0,8140	0,8157	0,8157	0,8157
	1500	0,7679	0,8089	0,8191	0,8157	0,8140	0,8140	0,8157	0,8157	0,8123

Taula 20. Precisions reals del Bagging en l'assignatura Informàtica

35	99
15	437

Figura 22. Matriu de confusió del Bagging per Informàtica

Predicció a l'assignatura *Materials*

S'aplica el mètode d'arbre de decisió en la predicció de l'aprovat de l'assignatura *Materials*. Les precisions calculades dels diferents models en funció dels paràmetres *n_estimators* i *max_samples* es troben en la *Taula 21*.

El model que inicialment obté major precisió és aquell que pren el màxim nombre de dades: tant el màxim nombre d'estimadors com el màxim nombre de mostres per estimador. La precisió obtinguda és de 8079.

Un cop es validen els models sobre les dades de *testing*, la màxima precisió, de 0,6724, es troba en el model que pren 50 estimadors i 250 mostres per estimador. Aquest model inicialment tenia una precisió de 0,7751, de forma que hi ha hagut una disminució de precisió de 0,1027 en la validació respecte la construcció del model.

Si prenem com a model definitiu el que obté major precisió en el *testing*, obtenim la matriu de confusió que es mostra a la *Figura 23*. De forma semblant en la predicció en altres assignatures, la precisió en la predicció del suspès és molt baixa (concretament de 0,1462) i la precisió de l'aprovat molt alta (pren un valor de 0,9632).

En l'assignatura de *Materials*, el model *Bagging* obtingut té una precisió (**0,6724**) molt menor a les obtingudes en les assignatures predites fins ara. La seva precisió en predir l'aprovat és alta però és molt baixa en el suspès. No es considera que el model sigui bon predictor sobre l'assignatura de *Materials*.

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174
20	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174
50	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174	0,7174
100	0,7668	0,7174	0,7174	0,7612	0,7568	0,7529	0,7585	0,7535	0,7557
150	0,7096	0,7674	0,7629	0,7657	0,7746	0,7679	0,7668	0,7690	0,7668
200	0,7468	0,7407	0,7418	0,7624	0,7718	0,7729	0,7707	0,7751	0,7718
250	0,7524	0,7662	0,7696	0,7746	0,7712	0,7751	0,7751	0,7707	0,7757
300	0,7451	0,7562	0,7668	0,7762	0,7785	0,7807	0,7829	0,7762	0,7768
500	0,7274	0,7735	0,7762	0,7823	0,7801	0,7768	0,7879	0,7840	0,7801
1000	0,7546	0,7868	0,7885	0,7934	0,7940	0,8001	0,7990	0,7996	0,7951
1500	0,7701	0,7968	0,7879	0,8023	0,8023	0,8040	0,8051	0,7973	**0,8079**

Taula 21. Precisions inicials del Bagging en l'assignatura Materials

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563
20	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563
50	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563	0,5563
100	0,6502	0,6587	0,6451	0,6399	0,6519	0,6706	0,6706	0,6672	0,6655
150	0,6485	0,6638	0,6638	0,6468	0,6672	0,6655	0,6553	0,6587	0,6672
200	0,6399	0,6587	0,6621	0,6655	0,6621	0,6451	0,6587	0,6536	0,6604
250	0,6485	0,6570	0,6655	0,6416	0,6536	0,6519	**0,6724**	0,6587	0,6536
300	0,6502	0,6263	0,6638	0,6570	0,6621	0,6519	0,6365	0,6553	0,6638
500	0,6655	0,6604	0,6689	0,6672	0,6587	0,6502	0,6655	0,6621	0,6638
1000	0,6195	0,6451	0,6570	0,6758	0,6553	0,6655	0,6706	0,6587	0,6689
1500	0,6416	0,6416	0,6724	0,6655	0,6706	0,6655	0,6672	0,6724	0,6655

Taula 22. Precisions reals del Bagging en l'assignatura Materials

38	222
12	314

Figura 23. Matriu de confusió del Bagging per Materials

Predicció a l'assignatura *Mecànica*

Procedim a l'aplicació de la tècnica *Bagging* per predir l'aprovat o suspès de l'assignatura *Mecànica* combinant diversos valors de paràmetres. Les precisions dels models es troben a la *Taula 23.*

Igual que en l'assignatura *Materials*, les precisions dels models són baixes quan es construeixen (no s'ajusten bé a les dades de *training*). Conseqüentment, les precisions dels models aplicats a les dades noves també seran baixes. En l'aplicació del model sobre les dades de *training*, la major precisió es troba pels valors màxims de *n_estimators* i *max_samples*.

Les precisions obtingudes en aplicar els models a les dades de *testing* es troben a la *Taula 22*. Tal i com s'havia previst ja, les precisions en general són molt baixes: el màxim valor és de 0,6724 i el mínim de 0,5563. Prenent com a model definitiu aquell que prediu millor les dades de *testing* (*n_estimators*=50, *max_samples*=250), es construeix la matriu de confusió de la *Figura 24*.

La matriu de confusió expressa que la precisió en la predicció del suspès és de 0,1558 mentre que la predicció de l'aprovat és de 0,9606. Torna a haver una diferència molt gran entre precisions, sent molt major la obtinguda predient el suspès.

Després d'analitzar els resultats, no es considera que el model actuï bé sobre l'assignatura *Mecànica* al tenir una precisió baixa (**0,6724**), a més de tenir una molt mala predicció del suspès.

		n_estimators								
		1	3	5	10	15	25	50	75	100
max_samples	10	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164
	20	0,4836	0,4836	0,5164	0,5164	0,4836	0,4836	0,5164	0,5164	0,5164
	50	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164	0,5164
	100	0,7285	0,7246	0,7168	0,7263	0,7285	0,7224	0,7252	0,7285	0,7301
	150	0,7285	0,7368	0,7290	0,7279	0,7290	0,7335	0,7335	0,7368	0,7346
	200	0,6996	0,7202	0,7324	0,7446	0,7340	0,7313	0,7390	0,7401	0,7413
	250	0,6996	0,7318	0,7396	0,7368	0,7401	0,7474	0,7507	0,7474	0,7424
	300	0,6713	0,7418	0,7435	0,7507	0,7479	0,7446	0,7590	0,7529	0,7463
	500	0,7268	0,7440	0,7562	0,7718	0,7512	0,7679	0,7646	0,7662	0,7607
	1000	0,7590	0,7662	0,7696	0,7840	0,7779	0,7779	0,7873	0,7868	0,7840
	1500	0,7474	0,7779	0,7779	0,7846	0,7912	0,7901	0,7896	0,7868	**0,7929**

Taula 23. Precisions inicials del Bagging en l'assignatura Mecànica

		n_estimators								
		1	3	5	10	15	25	50	75	100
max_samples	10	0,3942	0,3942	0,6058	0,3942	0,3942	0,3942	0,3942	0,6058	0,6058
	20	0,3942	0,3942	0,3942	0,3942	0,3942	0,3942	0,6058	0,3942	0,3942
	50	0,6058	0,3942	0,3942	0,3942	0,3942	0,3942	0,3942	0,3942	0,6058
	100	0,6314	0,6416	0,6314	0,6229	0,6280	0,6485	0,6348	0,6297	0,6314
	150	0,5836	0,6109	0,6331	0,6502	0,6195	0,6246	0,6280	0,6399	0,6314
	200	0,6809	0,6502	0,6399	0,6331	0,6314	0,6587	0,6416	0,6451	0,6399
	250	0,6263	0,6724	0,6263	0,6331	0,6399	0,6416	0,6297	0,6399	0,6485
	300	0,6314	0,6024	0,6229	0,6468	0,6433	0,6280	0,6587	0,6416	0,6399
	500	0,6536	0,6280	0,6263	0,6689	0,6587	0,6485	0,6502	0,6553	0,6468
	1000	0,6638	0,6468	0,6519	0,6536	0,6468	0,6433	0,6451	0,6672	0,6553
	1500	0,6160	0,6536	0,6587	0,6433	0,6399	0,6570	**0,6689**	0,6519	0,6553

Taula 24. Precisions reals del Bagging en l'assignatura Mecànica

36	195
14	341

Figura 24. Matriu de confusió del Bagging per Mecànica

Predicció a l'assignatura *Mètodes numèrics*

S'analitza el comportament de la tècnica *Bagging* sobre la predicció de la qualificació en l'última assignatura restant: *Mètodes numèrics.* Les precisions obtingudes en utilitzar els models sobre les dades de *training* es troben a la *Taula 25*.

En aquest cas s'observa que inicialment les precisions dels models són elevades, s'ajusten bé a les dades de *training*. La màxima precisió, de 0,916, s'obté en valors elevats de *n_estimators* i *max_samples*.

A continuació s'apliquen els models sobre les dades de *testing* i s'obtenen les precisions de la *Taula 26*. Ara la màxima precisió es troba pel model que pren major nombre d'estimadors i major nombre mostres per estimador, la qual és de 0,8003. El model havia obtingut una precisió de 0,9162, sent 0,1159 major respecte a l'obtinguda amb les dades de *testing*.

Prenent com a model definitiu el que obté màxima precisió en el *testing*, s'obté la matriu de confusió de la *Figura 24*. En ella s'indica que la precisió en la predicció de l'aprovat és de 0,9734 mentre que en el suspès és de 0,2815. Un cop més s'obté una precisió molt major predient l'aprovat.

Es pot concloure que, tot i obtenir una mala predicció del suspès en l'assignatura *Mètodes numèrics*, el model es pot prendre vàlid per la predicció al tenir una precisió de **0,8003**. Això es deu a que el nombre d'aprovats en l'assignatura és molt major al nombre de suspesos.

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
20	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
50	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
100	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
150	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
200	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
250	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
300	0,8840	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006	0,9006
500	0,8884	0,9006	0,9089	0,9078	0,9062	0,9006	0,9128	0,9023	0,9006
1000	0,9084	0,9095	0,9106	0,9123	0,9150	0,9150	0,9162	0,9150	0,9145
1500	0,9001	0,9100	0,9123	0,9150	0,9150	0,9150	0,9123	**0,9167**	0,9162

Taula 25. Precisions inicials del Bagging en l'assignatura Mètodes numèrics

max_samples \ n_estimators	1	3	5	10	15	25	50	75	100
10	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
20	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
50	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
100	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
150	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
200	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
250	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
300	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696	0,7696
500	0,8208	0,8140	0,7884	0,7730	0,7713	0,7679	0,7696	0,7679	0,7696
1000	0,7816	0,7884	0,8123	0,7918	0,7799	0,7867	0,7765	0,7730	0,7867
1500	0,7850	0,7816	0,7935	0,7969	0,7850	0,7901	0,7833	0,7833	**0,8003**

Taula 26. Precisions reals del Bagging en l'assignatura Mètodes numèrics

38	97
12	439

Figura 25. Matriu de confusió del Bagging per Mètodes numèrics

4.2.3. Comparació entre mètodes predictius

L'arbre de decisió és un mètode de predicció que en principi presenta molta variabilitat. La variabilitat en un model de predicció es dóna quan petits canvis en les dades de *training* resulten en canvis molt significatius en el model generat. En el cas de l'arbre de decisió, una petita variació en el conjunt de dades amb el que es modela l'arbre pot resultar en un canvi del node arrel de l'arbre i, conseqüentment, un canvi en tots els següents nodes interns. És per aquest motiu que s'han analitzat també les mateixes dades mitjançant la tècnica combinada *Bagging*, una eina que permet reduir la variabilitat d'un mecanisme de predicció a partir de l'ús de diferents estimadors.

Mitjançant el contrast entre els resultats obtinguts en els dos mètodes de predicció es pot veure si aquesta suposada variabilitat s'ha disminuït i s'ha traduït en una millor precisió del model. Les dades dels models que s'han pres com definitius per l'arbre de decisió es troben a la *Taula 27* i les corresponents al *Bagging* a la *Taula 28*.

Comparant les dues taules, s'observa que la precisió obtinguda en cada assignatura no varia significativament entre models, sí que s'aprecia un lleuger augment de precisió en el mètode combinat però la diferència és mínima. Si la diferència entre models no és significativa, vol dir que els arbres de decisions seleccionats no presenten gaire variabilitat.

S'observa també que, en general, les precisions del *testing en* la predicció de l'aprovat o suspès en les diferents assignatures no són molt elevades, tan sols superen el 80% en ambdós mètodes en les assignatures *Informàtica* i *Equacions diferencials*. La major precisió d'aquestes assignatures en comparació a les altres es pot deure a què s'hagin mantingut més uniformes en el temari i sistema d'avaluació durant el període estudiat, permetent que els patrons que segueixen els estudiants en aprovar-les o suspendre-les en la primera convocatòria siguin semblants.

Si mirem també les precisions del *training*, ens adonem que aquestes tampoc són gaire elevades. Si els models no s'han pogut ajustar bé a les dades i no han pogut captar els patrons que les caracteritzen, pot ser que el problema es trobi en les dades i no en els models de predicció. Una observació rellevant en les prediccions mitjançant *Bagging*, és la seva poca precisió en la predicció del suspès. Aquest fet pot ser degut al desequilibri en la distribució de classes, on el nombre d'aprovat sols ser major al de suspesos, dificultant la discriminació de la classe minoritària en la construcció d'arbres.

	max_depth	n_samples_leaf	precisió a *training*	precisió a *testing*	diferència de precisió
ELECTRO	10	35	0,8101	0,7065	-0,1036
EQDIF	3	50	0,8151	0,8737	0,0586
INFO	3	25	0,8395	0,8089	-0,0306
MAT	5	35	0,7962	0,6741	-0,1221
MEC	7	35	0,7812	0,6843	-0,0969
METNUM	3	25	0,9150	0,7952	0,1198

Taula 27. Models seleccionats i resultats obtinguts per l'arbre de decisió

	n_estimators	max_samples	precisió a *training*	precisió a *testing*	diferència de precisió
ELECTRO	50	1000	0,8029	0,7184	-0,0845
EQDIF	1	10	0,7857	0,8891	0,1034
INFO	75	250	0,8318	0,8345	0,0002
MAT	50	250	0,7751	0,6724	-0,1027
MEC	50	1500	0,7896	0,6689	-0,1207
METNUM	100	1500	0,9162	0,8003	-0,1159

Taula 28. Models seleccionats i resultats obtinguts pel Bagging

En el *Bagging*, al predir arbres amb subconjunts de dades, por ser que l'efecte s'amplifiqui i sorgeixin subconjunts amb molt pocs suspesos. Una altra raó per la qual el *Bagging* no hagi millorat pot ser que en fer models amb petits conjunts de dades, les reduccions d'arbres hagin perjudicat la variabilitat que s'esperava generar.

Un altre dels motius pels quals es va voler aplicar un mètode combinat és la possible aparició del fenomen *overfitting* en l'ús de només un predictor. El camp '*diferència de precisió*' de la *Taula 27* i la *Taula 28* indica quina diferència hi ha entre les precisions dels models aplicats a les dades de *training* i les precisions quan s'apliquen al *testing*.

Si hi hagués un sobreajustament de l'arbre de decisió en les dades, la diferència de precisió seria elevada. S'observa que la diferència de precisió en l'arbre de decisió és similar al cas del *Bagging*. Per tant, no es considera que en els arbres seleccionats s'hagi produït un *overfitting*.

A partir de l'anàlisi dels resultats obtinguts, es comprova que els factors que podien afectar a la precisió de l'arbre de decisió, la variabilitat i *l'overfitting,* no es presenten en els models seleccionats com a definitius. La baixa precisió dels models no es considera un problema en l'ajustament dels mecanismes de predicció sinó que es justifica a partir de les dades, les quals són poc representatives per ser predites.

Al final del treball es presenten quins poden ser els motius pels quals les dades no són representatives per la predicció i alternatives d'estudi que poden compensar aquesta manca d'informació.

4. PRESSUPOST

Els costos atribuïbles al present treball són baixos en comparació amb altres projectes de gran escala. El cost total es pot desglossar en costos de personal i costos d'infraestructura. A continuació es realitzen els càlculs necessaris per determinar el cost final del projecte. Tots els costs descrits es troben representats a la *Taula 29*.

Costs de personal

Els costs de personal fan referència al preu de treball per part d'un analista en el desenvolupament del projecte. El treball realitzat es pot dividir en tres components principals: investigació, anàlisi i presentació. El preu per hora comptabilitzat és diferent per cadascuna de les tres parts degut a la diferència de dificultat entre elles.

La investigació consisteix bàsicament en la fase de familiarització del problema i la metodologia, adquirint els coneixements necessàries per dur a terme el projecte. A l'anàlisi s'executen les tasques pròpies del treball requerides per la resolució del problema. Per últim, és necessari dedicar un temps de presentació de resultats i conclusions, a més de la documentació del procediment realitzat.

Costs d'infraestructura

El cost d'infraestructura es compon d'una part de recursos informàtics i una part de material d'oficina.

Les despeses a tenir en compte en recursos informàtics són tan sols les corresponents a la utilització d'un ordinador. Tot el programari utilitzat en l'estudi és lliure i de codi obert, de manera que no comporta cap cost.

S'ha emprat un ordinador portàtil valorat en 1200€ i es considera un cost de manteniment del 10% anual del seu preu d'adquisició per un ús de 1200 hores anuals. Establint un ús d'ordinador del 95% en el total d'hores de realització del projecte, el cost de manteniment és:

$$280h \cdot 0{,}95 \cdot \frac{1200€ \cdot 0{,}1}{1200h} = 26{,}60\ €$$

A més del cost de manteniment, en el còmput del cost de l'ordinador cal tenir en compte el càlcul de la seva amortització. Considerant un ús de 48 setmanes a l'any durant 3 anys des del moment de compra. L'ordinador ha sigut utilitzat en el treball durant 5

PERSONAL			
Concepte	**Preu per hora**	**Hores**	**Cost total**
Investigació	30€/h	50h	1.500€
Anàlisi	40€/h	200h	8.000€
Presentació	25€/h	30h	750€
INFRAESTRUCTURA			
Concepte			**Cost total**
Recursos informàtics			184,93€
Material d'oficina (fulls)			20€
COST TOTAL			10.454,93€

Taula 29. Costos del projecte

mesos, és a dir, 22 setmanes. Si es descompta un dia de descans per setmana, l'ús real és de 19 setmanes.

$$19\ set \cdot \frac{\frac{1200€}{3anys}}{48\ set} = 158{,}33\ €$$

Per tant, l'ús de recursos informàtics comporta un cost total de:

$$26{,}60€ + 158{,}33€ = \mathbf{184{,}93}€$$

En la part de material d'oficina només es computen els costs derivats a l'ús de recursos com paper i bolígrafs durant les diferents fases del projecte, juntament amb els possibles residus generats. Es considera un cost aproximat de **20€.**

Reunint tots els costs derivats de la realització del treball, el pressupost del projecte és d'un total de 10.454,93€, tal i com es mostra a la *Taula 29*.

5. IMPACTE AMBIENTAL

L'impacte ambiental produït pel projecte és mínim donat que, al tractar-se d'un treball de caire informàtic, les tasques han sigut dutes a terme mitjançant un ordinador i no s'ha generat cap residu.

El residu generat en forma de paper degut a l'ús de material d'oficina no es contempla en aquest apartat ja que ha sigut avaluat ja econòmicament en la secció de *Pressupost*.

L'únic element que es pot tenir en compte és l'ús d'energia elèctrica per l'alimentació de l'ordinador i l'encaminador, conegut com a *router*, que proporciona connexió sense fils a Internet. L'enllumenat del lloc de treball es pot també considerar, tot i que la majoria de tasques s'han realitzat aprofitant llum natural i la il·luminació utilitzada és necessària també quan no s'està treballant.

6. PLANIFICACIÓ DEL PROJECTE

A continuació es mostra en un diagrama *Gantt* la planificació de les diferents tasques que comprèn la realització del projecte, repartides en les fases d'inici del projecte, la preparació de dades, el modelatge i validació de predictors, i finalment, la confecció pròpia de la memòria i la posterior presentació del projecte.

		ACTIVITATS	set-18	oct-18	nov-18	des-18	gen-19
INICI DEL PROJECTE	1	Definició de la metodologia a seguir					
	2	Instal·lació de les diferents eines					
	3	Familiarització amb la llibreria Pandas					
PREPARACIÓ DE DADES	4	Neteja de dades					
	5	Transformació de dades					
MODELATGE I VALIDACIÓ	6	Estudi dels algorismes de predicció					
	7	Automatització del modelatge d'algorismes					
	8	Aplicació d'algorismes i validació					
	9	Anàlisi de resultats					
CONFECCIÓ DE LA MEMÒRIA	10	Redacció de la memòria					
	11	Pressupost					
	12	Estudi de l'impacte ambiental					
	13	Conclusions					
PRESENTACIÓ	14	Presentació del projecte					

7. CONCLUSIONS

Després de realitzar el present treball, es considera que s'han cobert tots els objectius marcats inicialment. En tot el procés d'estudi s'ha seguit una metodologia CRISP adaptada a les característiques del treball, definint les tasques a realitzar en cada fase i garantint la possibilitat de que sigui replicada a partir de la documentació proporcionada.

En l'anàlisi de resultats, s'ha pogut estudiar la precisió dels models de predicció emprats, l'arbre de decisió i el mecanisme *Bagging*, en funció de diferents paràmetres. La validació dels models s'ha dut a terme de forma sistemàtica i sota mateixes condicions, per tal que els resultats obtinguts puguin ser contrastats. S'ha comprovat que la definició dels paràmetres permet evitar el sobreajustament dels models sobre les dades. Les precisions obtingudes en l'arbre de decisió en la predicció del suspès han sigut baixes i l'aplicació del mètode combinat no ha pogut millorar-les significativament. Per aquest motiu, es considera que les dades utilitzades no són prou representatives per poder ser predites, fet que pot ser degut a diversos motius.

Les dades analitzades cobreixen el període de 2010-2017, on l'any 2010 va ser just quan es va introduir el nou pla d'estudis del grau. Des de llavors, totes les assignatures han estat evolucionant fins arribar a una certa estabilitat en quant a temari i avaluació. Les dades de *training* emprades corresponen al període 2010-2015, temps en què el nou pla d'estudis començava a implementar-se i podien haver diversos canvis en els programes acadèmics.

Un altre motiu pot ser que l'ús de dades purament acadèmiques corresponents a l'ETSEIB no siguin suficients per poder predir qualificacions d'assignatures. La introducció de dades externes, com poden ser dades relacionades amb l'àmbit familiar o amb el rendiment acadèmic abans d'entrar a l'escola, poden ajudar a que els models s'ajustin més a les dades i conseqüentment millorar la precisió de la predicció.

Tot i no haver obtingut unes precisions elevades en la predicció, s'ha complert l'objectiu principal, el qual és l'estudi del rendiment de les tècniques de mineria de dades en la predicció de l'aprovat o suspès en les assignatures corresponents al Q3.

Conclusions personals

Personalment, es vol destacar també els diferents coneixements que ha aportat la realització d'aquest projecte. S'han pogut aplicar les nocions de programació en llenguatge *Python*, adquirides en les assignatures d'informàtica del grau, sobre un cas pràctic. S'ha familiaritzat amb la llibreria *Pandas* i s'ha conegut les facilitats que proporciona en nombroses funcions. Però, per sobre de tot, s'ha tingut coneixement de la importància de la mineria de dades, la gran varietat d'aplicacions que té i la quantitat de nova informació que és capaç d'extreure d'un conjunt de dades.

Treball futur

Un cop realitzat el treball a partir dels resultats obtinguts es poden plantejar alternatives d'anàlisi que es poden dur a terme amb la mateixa metodologia general.

En les dades inicials que es disposen, s'inclouen dades de codis postals d'estudiants, corresponents a codis postals familiars i codis postals dels centres d'educació d'on provenen. La relació entre el codi postal familiar amb la renda econòmica o el codi postal del centre d'educació amb el barri que correspon són variables que es poden incloure com a variables de predicció.

A la part de preparació de dades, s'han definit noves variables de predicció a partir de les dades ja existents, les quals contenen informació que es troba de forma implícita dins les dades. La definició de noves variables pot ajudar a captar més patrons en el conjunt de dades.

En el treball la divisió de les dades en *training* i *testing* s'ha fet amb un any de tall del 2015. Es poden analitzar els resultats obtinguts amb diferents anys de tall per saber si hi ha variabilitats en les dades que no s'han pogut apreciar amb la divisió aplicada.

El percentatge d'aprovats i suspesos no és el mateix en cada assignatura. Hi ha assignatures on el nombre d'aprovats és major al de suspesos i el model, al tenir més dades d'aprovats, té més precisió en la predicció de l'aprovat. El mateix succeeix en el cas contrari, quan el nombre de suspesos és major al d'aprovats. Per intentar compensar la proporció d'aprovats i suspesos, es poden aplicar tècniques de *resampling*, on es repliquen dades dins d'un mateix conjunt. Cal tenir en compte que una mala aplicació del *resampling* pot resultar en un falsejament de dades.

BIBLIOGRAFIA

[1] Han, Jiawei et al. *Data Mining: Concepts and Techniques*. 3rd ed. Waltham: Elsevier, 2012. 673 p. ISBN 978-0-12-381479-1.

[2] Olson, David L. et al. *Advanced Data Mining Techniques.* Berlin: Springer, 2008. 169 p. ISBN 978-3-54-076916-3.

[3] Pyle, Dorian. *Data Preparation for Data Mining*. San Francisco: Morgan Kaufmann Publishers, 1999. 460 p. ISBN 978 -1-55-860529-9.

[4] Witten, Ian H. et al. *Data Mining: Practical Machine Learning Tools and Techniques*. 3rd ed. Burlington: Morgan Kaufmann Publishers, 2011. 617 p. ISBN 978-0-12-374856-0.

[5] *Diversos autors*. *KDNuggets.* Adreça web: https://www.kdnuggets.com

[6] Diversos autors. *Stackoverflow* (fòrum de desenvolupadors de codi). Adreça web: https://stackoverflow.com

[7] *Scikit-learn: Machine learning in Python* (documentació de la llibreria *scikit-learn*). Adreça web: https://scikit-learn.org/stable/

[8] *Python Data Analysis Library* (documentació de la llibreria *Pandas*. Adreça web: https://pandas.pydata.org

ANNEX

A1. Preparació de dades

```
import pandes as pd  #importa llibreria Pandas

preins = pd.read_excel("dadespersnomespreins.xlsx")
fasein = pd.read_excel("qfaseini.xlsx")
fasenoin = pd.read_excel("qfasenoini.xlsx") #lectura dels xlsx

preins.to_csv("preins.csv",index=False)
fasein.to_csv("fasein.csv",index=False)
fasenoin.to_csv("fasenoin.csv",index=False) #es passen els xlsx a csv

preins = pd.read_csv("preins.csv")
fasein = pd.read_csv("fasein.csv")
fasenoin = pd.read_csv("fasenoin.csv") #lectura dels csv

#-------------------------------------------------------------------------------#
#NETEJA DADES:

#ELIMINAR EXPEDIENTS QUE NO HAN SUPERAT FASE SELECTIVA
valors = set(fasenoin['CODI_EXPEDIENT'])
fasein['SIFASENOIN'] = fasein['CODI_EXPEDIENT'].isin(valors).astype(int) #columna 'SIFASENOIN'
indica si els codi expedient de fasein existeixen a fasenoin
fasein = fasein[fasein['SIFASENOIN'] == 1]
fasein = fasein.drop('SIFASENOIN', axis=1)#s'esborren expedients de fasein que no apareixen a fasenoin
#fusió dades_assig = fasein + fasenoin
dades_assig = pd.concat([fasein,fasenoin])

#ELIMINAR FILES DUPLICADES
dades_assig = dades_assig.drop_duplicates()

#ELIMINAR COLUMNA TIPUS ACCES (nomes pren valor 1)
preins = preins.drop(['TIPUS_ACCES'],axis=1)

#ELIMINAR EXPEDIENTS QUE NO CORRESPONEN AL GRAU
dades_assig = dades_assig[dades_assig.CODI_PROGRAMA == 752]
dades_assig = dades_assig.drop(columns=['CODI_PROGRAMA'])#s'esborra la columna
'CODI_PROGRAMA'

#ELIMINAR FILES DE CONV QUE NO SIGUIN DE Q1/Q2/Q3
#lectura de la taula amb assignatures de Q1,Q2,Q3
assign = pd.read_excel("assign.xlsx")
assign.to_csv("assign.csv",index=False)
assign = pd.read_csv("assign.csv")
#s'agafen les files corresponents a assignatures que apareixen a la taula
assign['codi']=assign['codi'].apply(lambda x:str(x))
sassign = set(assign['codi']) #set de codis de les assignatures desitjades
dades_assig['CODI_UPC_UD']=dades_assig['CODI_UPC_UD'].apply(lambda x:str(x))
dades_assig = dades_assig[dades_assig['CODI_UPC_UD'].isin(sassign)]

#CERCAR FILES AMB VALORS NULS
valorsnuls = dades_assig[dades_assig.isnull().any(axis=1)]

#fins ara dades_assig es DF amb una fila per expedient-convocatoria
#es vol passar a una fila per expedient
```

```
#TRANSFORMACIÓ DADES:

#CREACIÓ TAULA AMB UNA FILA PER EXPEDIENT
#columna 'CONV' indica el nº de convocatoria per cada codiexp-assignatura
dades_assig = dades_assig.sort_values(by=['CURS','QUAD'])
dades_assig['CONV'] = dades_assig.groupby(['CODI_EXPEDIENT','CODI_UPC_UD']).cumcount() + 1
#es substitueix CODIASSIG per NOMASSIG
dades_assig['CODI_UPC_UD'] = dades_assig['CODI_UPC_UD'].map(assign.set_index('codi')['nom'])
dades_assig = dades_assig.rename(columns={'CODI_UPC_UD': 'NOMASSIG'}) #es canvia el nom de la
columna
#creació columna CONV-NOMASSIG
dades_assig['ASSIGCONV'] = dades_assig[['NOMASSIG','CONV']].apply(lambda x:'-
'.join([str(x[0]),str(x[1])]),axis=1)
#es pivota el DF creant tassig: CODI_EXPEDIENT com a files, ASSIG-CONV com a columnes i
NOTES_DEF com a valors de la taula
tassig = dades_assig.pivot(index='CODI_EXPEDIENT', values='NOTA_NUM_DEF',
columns='ASSIGCONV')

#NOTES NUMÈRIQUES A NOTES QUALITATIVES
def qualificacio(nota):
    if nota<5:
        return 1
    elif 5 <= nota < 7:
        return 2
    elif 7 <= nota < 9:
        return 3
    elif 9<= nota <=10:
        return 4
    else:
        pass
tassig = tassig.applymap(qualificacio)

#dades_tot = tassig + preins
dades_tot = pd.merge(preins,tassig,on='CODI_EXPEDIENT',how='inner',left_index=True)
dades_tot = dades_tot.sort_values(by=['ANY_ACCES'])

#-------------------------------------------------------------------------------#
#ADDICIO NOVES COLUMNES:

#N_ASS_SUSP_FASEIN (nº d'assignatures suspeses per cada expedient a fasein)
#t = DF amb el nº de conv cursades de cada assig per expedient
dades_assig['CONVMAX'] =
dades_assig.groupby(['CODI_EXPEDIENT','NOMASSIG'])['CONV'].transform(max)
t = dades_assig.drop_duplicates(['CODI_EXPEDIENT','NOMASSIG','CONVMAX'])
t = t.pivot(index='CODI_EXPEDIENT',values='CONVMAX',columns='NOMASSIG')
#columna NASS_SUSP_FASEIN indica el nº d'ass suspeses a fasein
tin = t.drop(['ELECTRO','EQDIF','METNUM','INFO','MAT','MEC'], axis=1)#selecciona files de fasein nomes
tin['NASS_SUSP_FASEIN']= tin.apply(lambda row: sum(row[:]>1),axis=1)
#s'afegeix la columna NASS_SUSP_FASEIN a la taula principal dades_tot
tnass = tin[['NASS_SUSP_FASEIN']].copy()
dades_tot = pd.merge(dades_tot,tnass,on='CODI_EXPEDIENT',left_index=True)


#N_QUAD_FASEIN (nº de quadrimestres en passar la fasein)
#columna ANYQUAD = ANYQUAD de cada convocatòria
fasein['ANYQUAD']= fasein[['CURS','QUAD']].apply(lambda x: str(x[0])+str(x[1]),axis=1)
fasein['ANYQUAD']=fasein['ANYQUAD'].apply(lambda x:int(x)) #es passa a format int
```

```
#columnes PRIMERANYQUAD (primer quad cursat per assig) i ULTIMANYQUAD (últim quad cursat per assig)
fasein['PRIMERANYQUAD'] = fasein.groupby(['CODI_EXPEDIENT'])['ANYQUAD'].transform(min)
fasein['ULTIMANYQUAD'] = fasein.groupby(['CODI_EXPEDIENT'])['ANYQUAD'].transform(max)
#creació columna N_QUAD_FASEIN
fasein['PRIMERANYQUAD']=fasein['PRIMERANYQUAD'].apply(lambda x:str(x))
fasein['ULTIMANYQUAD']=fasein['ULTIMANYQUAD'].apply(lambda x:str(x) )#es passen a format str
fasein['N_ANY_FASEIN']=fasein[['PRIMERANYQUAD','ULTIMANYQUAD']].apply(lambda x:int(x[1][:4])-int(x[0][:4]),axis=1) #diferencia d'anys entre primerquad i ultimquad
fasein['N_DIFQUAD_FASEIN']=fasein[['PRIMERANYQUAD','ULTIMANYQUAD']].apply(lambda x:int(x[1][4])-int(x[0][4]),axis=1)#diferencia de quad entre primerquad i ultimquad (-1/0/1)
fasein['N_QUAD_FASEIN'] = fasein.apply(lambda fila:(fila['N_ANY_FASEIN']*2) if fila['N_DIFQUAD_FASEIN']<0 else (fila['N_ANY_FASEIN']+fila['N_DIFQUAD_FASEIN'])*2,axis=1)
#s'afegeix N_QUAD_FASEIN a la taula principal dades_tot
tnquad = fasein[['CODI_EXPEDIENT','N_QUAD_FASEIN']].copy()
dades_tot = pd.merge(dades_tot,tnquad,on='CODI_EXPEDIENT',how='inner',left_index=True)
dades_tot = dades_tot.drop_duplicates()

#NMITJ_FASEIN (nota mitjana fasein de cada exp comptant totes les convocatòries)
#creació NMITJ_FASEIN
tassig['NMITJ_FASEIN']=tassig.mean(axis=1)
#s'afegeix NMITJ_FASEIN a la taula principal dades_tot
tnmitj = tassig[['NMITJ_FASEIN']].copy()
dades_tot = pd.merge(dades_tot,tnmitj,on='CODI_EXPEDIENT',how='inner',left_index=True)
dades_tot = dades_tot.drop_duplicates()

#GRUP_ACCES (grup matriculat al primer quad)
#faseinprimergrup = DF amb la primera fila de cada exp i creació PRIMER_GRUP
fasein = fasein.reset_index(drop=True)
faseinprimergrup = fasein.dropna() #s'eliminen files amb missing values
faseinprimergrup = faseinprimergrup[~faseinprimergrup['GRUP_CLASSE'].isin(['CONV'])]#s'eliminen aquelles conv convalidades
faseinprimergrup = faseinprimergrup.groupby('CODI_EXPEDIENT').first() #es selecciona la primera fila on apareix cada exp, al estar ordenades cronològicament
faseinprimergrup=faseinprimergrup.rename(columns={'GRUP_CLASSE': 'PRIMER_GRUP'})#els grups de classe obtinguts son els primers de cada exp
#s'afegeix PRIMER_GRUP a la taula principal dades_tot
tpgrup = faseinprimergrup[['PRIMER_GRUP']].copy()
dades_tot = pd.merge(dades_tot,tpgrup,on='CODI_EXPEDIENT',how='inner',left_index=True)
#creació GRUP_ACCES = PRIMER_GRUP acabat en 0
dades_tot['PRIMER_GRUP']=dades_tot['PRIMER_GRUP'].apply(lambda x:int(x))
dades_tot['GRUP_ACCES'] = dades_tot.apply(lambda fila:fila['PRIMER_GRUP'] if fila['PRIMER_GRUP']%10==0 else(fila['PRIMER_GRUP']-2 if fila['PRIMER_GRUP']%2==0 else (fila['PRIMER_GRUP']-3 if fila['PRIMER_GRUP']%3==0 else fila['PRIMER_GRUP']-1)),axis=1)
#s'esborren els GRUP_ACCES > 100
dades_tot = dades_tot[dades_tot.GRUP_ACCES <= 100]
dades_tot = dades_tot.reset_index(drop=True)


#passar qualificacio a 'Aprovat' o 'Suspes'
def notabinaria(nota):
    if nota<5:
        return 0
    else:
        return 1
```

```
#passar SEXE a valor categòric binari (0=H,1=D)
def sexe(sexe):
    if sexe == 'H':
        return 0
    elif sexe == 'D':
        return 1

dades_tot['SEXE'] = dades_tot['SEXE'].apply(lambda x: sexe(x))
```

A2. Modelatge i validació

```
#creacio X(columnes per construir model prediccio) i Y (columna a predir)
def XY(dades_tot,columnaY): #dades_tot es un DF
    assignfasenoin = assign[assign['quad'] == 3]
    assignfasenoin = list(assignfasenoin['nom'])#noms assignatures fasenoin
    colfasenoin=[]
    for columna in list(dades_tot.columns.values): #es seleccionen aquelles columnes que es volen treure
        if columna[:-2] in assignfasenoin:
            colfasenoin.append(columna)
    X = dades_tot.copy()
    X = X.drop(colfasenoin,axis=1)#es crea X eliminant les columnes seleccionades
    Y = dades_tot.copy()
    Y = dades_tot[[columnaY + '-1','CODI_EXPEDIENT']]
    Y.set_index('CODI_EXPEDIENT',inplace=True)
    return X,Y

#-----------------------------------------------------------------------------#
#HOLDOUT
def traintest(dades_tot,tallany): #dades_tot es un DF, tallany es un int
    tanys = dades_tot[['ANY_ACCES']]
    tanys = tanys.drop_duplicates()
    anys = list(tanys['ANY_ACCES']) #llista d'anys
    anys_train = []
    anys_test = []
    for any in anys:
        if any < tallany:
            anys_train.append(any)
        else:
            anys_test.append(any)
    X_train = X[X['ANY_ACCES'].isin(anys_train)]
    X_test = X[X['ANY_ACCES'].isin(anys_test)]
    Y_train = Y[Y.index.isin(X_train['CODI_EXPEDIENT'])]
    Y_test = Y[Y.index.isin(X_test['CODI_EXPEDIENT'])]
    X_train.set_index('CODI_EXPEDIENT',inplace=True)
    X_test.set_index('CODI_EXPEDIENT',inplace=True)
    return X_train,X_test,Y_train,Y_test
#-----------------------------------------------------------------------------#
#IMPLEMENTACIO DECISION TREE
def decisiontree(X_train,Y_train,max_depth,min_samples_leaf):
    from sklearn import tree
    clf = tree.DecisionTreeClassifier(min_samples_leaf=min_samples_leaf,max_depth=max_depth)
    clf = clf.fit(X_train,Y_train)
    return clf

def dt_diagram(clf,Xtrain):
    from sklearn import tree
    import graphviz
```

```
    import pydot

    Ynames = ['Suspès','Aprovat','Notable','Excel·lent']
    outfile = tree.export_graphviz(clf, out_file='dt_diagram.dot',feature_names
=list(Xtrain.columns.values),class_names=Ynames)
    (graph, )=pydot.graph_from_dot_file('dt_diagram.dot')
    graph.write_png('dt_diagram.png')


def dt_testing(clf,X_test,Y_test):
    from sklearn.metrics import accuracy_score
    from sklearn.metrics import confusion_matrix

    Y_pred_test = clf.predict(X_test)
    accuracy_score = accuracy_score(Y_test,Y_pred_test)
    return confusion_matrix(Y_test, Y_pred_test),accuracy_score


    df = pd.DataFrame(columns=[3,5,7,10,12,14,16,18,20],index=[1,3,5,10,15,20,25,30,35,40,50])

    lmaxdepth=[3,5,7,10,12,14,16,18,20]
    lminsamplesleaf=[1,3,5,10,15,20,25,30,35,40,50]

    X,Y = XY(dades_tot,'ASSIGN')
    Y = Y.applymap(notabinaria)
    Xtrain,Xtest,Ytrain,Ytest = traintest(dades_tot,2015)
    Xtrain = Xtrain.drop(columns=['ANY_ACCES'])
    Xtest = Xtest.drop(columns=['ANY_ACCES'])

    for maxdepth in lmaxdepth:
        for minsamplesleaf in lminsamplesleaf:
            clf = decisiontree(Xtrain,Ytrain,maxdepth,minsamplesleaf)
            CM,accuracy = dt_testing(clf,Xtrain,Ytrain)
            df.at[minsamplesleaf,maxdepth] = accuracy

#-------------------------------------------------------------------------------#
#IMPLEMENTACIO BAGGING
def bagging(X_train,Y_train,n_estimators,max_samples,assign):
    from sklearn.ensemble import BaggingClassifier
    from sklearn import tree
    bgg = BaggingClassifier(tree.DecisionTreeClassifier(min_samples_leaf =
35,max_depth=5),n_estimators = n_estimators,max_samples=max_samples)
    bgg = bgg.fit(X_train,Y_train[assign+'-1'])
    return bgg

def dt_testing(clf,X_test,Y_test):
    from sklearn.metrics import accuracy_score
    from sklearn.metrics import confusion_matrix

    Y_pred_test = clf.predict(X_test)
    accuracy_score = accuracy_score(Y_test,Y_pred_test)
    return confusion_matrix(Y_test, Y_pred_test),accuracy_score

    df=pd.DataFrame(columns=[1,3,5,10,15,25,50,75,100],index=[10,20,50,100,150,200,250,300,500,1000,
1500])

    lnestimators=[1,3,5,10,15,25,50,75,100]
    lmaxsamples=[10,20,50,100,150,200,250,300,500,1000,1500]
```

```
X,Y = XY(dades_tot, 'ASSIGN')
Y = Y.applymap(notabinaria)
Xtrain,Xtest,Ytrain,Ytest = traintest(dades_tot,2015)
Xtrain = Xtrain.drop(columns=['ANY_ACCES'])
Xtest = Xtest.drop(columns=['ANY_ACCES'])

for nestimators in lnestimators:
    for maxsamples in lmaxsamples:
        bgg = bagging(Xtrain,Ytrain,nestimators,maxsamples, 'ASSIGN')
        CM,accuracy = dt_testing(bgg,Xtest,Ytest)
        df.at[maxsamples,nestimators] = accuracy
```